Artificial Intelligence 175 (2011) 1–24

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

John McCarthy's legacy

Leora Morgenstern [a,*], Sheila A. McIlraith [b]

[a] SAIC, Advanced Systems and Concepts, Arlington, VA, USA
[b] Department of Computer Science, University of Toronto, Toronto, Canada

ARTICLE INFO

Article history:
Received 31 May 2010
Received in revised form 9 November 2010
Accepted 9 November 2010

Keywords:
Knowledge representation
Reasoning
Reasoning about actions
Nonmonotonic reasoning
Reasoning about knowledge
Domain axiomatizations
Elaboration tolerance

ABSTRACT

This special issue is dedicated to John McCarthy, founding father of Artificial Intelligence. It contains a collection of recent contributions to the field of knowledge representation and reasoning, a field that McCarthy founded and that has been a main focus of his research during the last half century. In this introductory article, we survey some of McCarthy's major contributions to the field of knowledge representation and reasoning, and situate the papers in this special issue in the context of McCarthy's previous work.

1. Introduction

This special issue is dedicated to John McCarthy, leading figure in the early days of computer science and founding father of Artificial Intelligence (AI). The papers collected here are part of the intellectual tradition rooted in his 1959 paper "Programs with Common Sense" [102], the tradition known within the Artificial Intelligence community as the logic-based approach to AI.

In that paper, McCarthy described his conception of an intelligent machine as an *advice taker* and reasoner. The machine would contain a set of sentences in a formal logical language, such as the first-order predicate calculus, representing the knowledge that it already had; it would be able to get information from other sources, and also store this information as sentences of a formal language; and it would include a mechanism for reasoning with, and making inferences from, these sentences. He envisioned the reasoning mechanism as entirely general; any domain-specific heuristics would be specified declaratively to the program. By means of this general reasoning mechanism, the machine would be able to reason from its knowledge, mirroring the ability of intelligent beings.

What was so innovative about McCarthy's idea? Certainly, the connection between intelligent thought and formal logic had been made long before. The attempt to formalize human thought within a logical language, and in particular, to furnish a concrete method for determining whether arguments were valid, dates as far back as Aristotle [3] and the Stoics [64], and continued through Roman and medieval times [11,64]. More than three centuries ago, in what was perhaps an early articulation of the logicist enterprise, Leibniz [79] conceived of a logical language in which basic concepts of human thought could be expressed, and a mechanizable procedure for reasoning with these basic concepts. Similarly, Frege [37], in his foundational work on modern logic, explicitly appealed to the formal representation of thought. But the work of Frege,

* Corresponding author.
E-mail addresses: leora.morgenstern@saic.com (L. Morgenstern), sheila@cs.toronto.edu (S.A. McIlraith).

0004-3702/$ – see front matter © 2010 Published by Elsevier B.V.
doi:10.1016/j.artint.2010.11.003

and of the other founders of modern logic, such as Whitehead and Russell [179] and Gödel [49,50], focused on formalizing *mathematical* thought, rather than the everyday reasoning in which people typically engage.

It was McCarthy who pioneered the attempt to formalize ordinary commonsense reasoning within the formal languages that had been used to represent mathematical truths.[1] McCarthy realized that commonsense reasoning was ubiquitous in intelligent behavior, and that therefore, a truly intelligent machine had to be capable of performing commonsense reasoning. He contended that the facts of commonsense knowledge could be expressed in formal logic in much the same way as the truths of mathematics; and that the process for reasoning with these facts would be quite similar to general methods used for proving theorems of mathematics. He wrote:

> There is an immediate deduction routine which when given a set of premises will deduce a set of immediate conclusions. Initially, the immediate deduction routine will simply write down all one-step consequences of the premises. Later, this may be elaborated so that the routine will produce some other conclusions which may be of interest.

When McCarthy wrote "Programs with Common Sense," this was a far reaching assumption. In the late 1950s, techniques for automated theorem proving — an obvious prerequisite for any implementation of the advice taker — were in their infancy; general efficient methods did not become available until several years later [149,150,99].[2]

More fundamentally, even the simple problem discussed in "Programs with Common Sense," that of planning to get to the airport, demonstrated that representing and reasoning with knowledge was a complex and difficult enterprise. In that paper, and in the course of many subsequent papers, McCarthy noted many of the difficult representational and reasoning issues that arise in formalizing seemingly straightforward commonsense problems. McCarthy's research has been devoted to solving these problems, and he has inspired several generations of AI researchers to work on these problems as well. The papers in this special issue all arise from research on the problems that he first noted.

This article surveys the major problems of knowledge representation and reasoning with which McCarthy has been most concerned, as well as some of the research in the AI community inspired by McCarthy's work, and discusses the contributions to this special issue in light of this research. The article is organized as follows: Sections 2, 3, and 4 discuss three of the areas most central to McCarthy's research: reasoning about action, reasoning about knowledge, and nonmonotonic reasoning. Section 5 surveys some of McCarthy's later and lesser-known research interests, including elaboration tolerance and domain axiomatizations. Section 6 summarizes each of the contributions to this special issue, and discusses its connection with McCarthy's research.

The article is not meant to cover all aspects of McCarthy's work in knowledge representation and reasoning. A collection of his major papers, together with a discussion of each of his papers, can be found in [92]. The article also does not address many of McCarthy's extraordinary achievements: his leading role in the organization of the Dartmouth conference of 1956; his creation of the Stanford AI laboratory; his role in the development of time sharing; his Turing Award, awarded in 1971 for his contributions to Artificial Intelligence; and his National Medal of Science, awarded in 1991. A survey of McCarthy's life and achievements is in [58].

2. Representing and reasoning about action

In contemplating the creation of an *advice taker* in the late 1950s, McCarthy recognized the need to develop a logical formalism that facilitated axiomatizing actions and their effects on the world. This realization led to one of his greatest contributions to Artificial Intelligence, the development of the situation calculus [103,121].[3] The situation calculus is perhaps the oldest special-purpose knowledge representation formalism. McCarthy's introduction of the situation calculus in 1963 [103] and his later work with Hayes in 1969 [121] have provided the formal foundation and inspiration for decades of research on representing and reasoning about dynamical systems in logic. It has led to the identification and study of fundamental problems such as the frame, qualification, and ramification problems, to the development of a handful of related logical formalisms for reasoning about action and change, and to the characterization and study of a variety of reasoning tasks associated with dynamical systems, most notably temporal projection and planning. In what follows we describe McCarthy's seminal contributions to the problem of representing and reasoning about action, and outline some of the later works that were influenced by his vision.

[1] The term *logicism* has traditionally referred to the school of thought, most explicitly expressed in [179], that mathematics can be reduced to formal logic; in the AI community, *logicism* refers to the school of thought, founded by McCarthy, that intelligent reasoning can be expressed within formal logic.

[2] There was, however, a burgeoning excitement about the possibility of creating a machine that could prove non-trivial theorems [20]. In 1954, Martin Davis had programmed an early (vacuum-tube) computer to prove additive theorems of Presburger arithmetic [18]. Shortly thereafter, there were several efforts, due to Newell, Shaw, and Simon [134] (cited in and a likely inspiration for "Programs with Common Sense"), to Wang [178], and others [46] to create automated procedures for proving theorems in the *Principia Mathematica*; and due to Gelernter [39] for proving theorems in geometry. In 1957, Abraham Robinson [148] suggested using Skolem functions and Herbrand's theorem as tools in developing general-purpose theorem provers. This was to lead to the development of the Davis–Putnam procedure [22,23] and eventually to the development of the efficient Davis–Putnam–Logemann–Loveland procedure [21] and of J. Alan Robinson's full-fledged efficient resolution [150].

[3] The situation calculus was not so named until [121, Section 2].

2.1. McCarthy's situation calculus

The example problem that McCarthy presented in his 1959 article on "Programs with Common Sense" is of an intelligent agent that plans to get to the airport. McCarthy considered the different types of commonsense facts that would be necessary for an agent to reason that it could successfully plan to arrive at the airport.

First, it would need problem-specific facts about the world such as "x is at y," or that "location x is walkable," e.g.,

$$at(I, desk)$$

$$at(airport, county)$$

$$walkable(home)$$

$$drivable(county)$$

Next, it would need information about the actions that it was considering performing. For example, there could be rules associated with the action of *going* — described as $go(x, y, z)$ ("I go from x to y using method z") — that specify the feasibility of such actions, i.e., the conditions under which it is possible to go from x to y by walking or driving. For example, the following axioms give sufficient conditions for the feasibility of some instances of the *go* action:

$$walkable(x), at(y, x), at(z, x), at(I, y) \to can\big(go(y, z, walking)\big)$$

$$drivable(x), at(y, x), at(z, x), at(car, y), at(I, car) \to can\big(go(y, z, driving)\big)$$

In addition, McCarthy identified rules concerning the effects of going from one location to another. For example, the following axiom expresses the fact — obvious to us, but not to a computer — that when an agent has gone to some location, he winds up being at that location:

$$did\big(go(x, y, z)\big) \to at(I, y)$$

McCarthy further identified a set of problem-independent logical formulas that would lead to the entailment of a particular fact about the world. In addition, he made some preliminary remarks about how these premises would be collected and how the deduction would proceed once they were found.

Besides being visionary in its conception of how a machine could reason with and about commonsense knowledge, this early paper also made at least two important technical contributions. First, it identified three separate categories of domain-specific axioms that are still exploited by most action formalisms in use today: facts about the world, rules that state conditions for the feasibility of actions in the world, and rules that state actions' effects on the world. Second. it specified that these axioms would be used by a deductive reasoner to reason about dynamical systems.

By 1963, McCarthy had further elaborated his ideas. In a technical note entitled "Situations, Actions, and Causal Laws" [103], McCarthy first introduced the basic entities of the situation calculus: *situations*, *actions*, and *fluents*. To McCarthy, and later, McCarthy and Hayes, a situation was a complete state of affairs at some instant in time. McCarthy's conception of a situation differs in this sense from the later axiomatization of the situation calculus by Reiter [146]. For Reiter, a situation was an action history — the finite sequence of actions which had been performed starting in the distinguished initial situation, S_0. Despite this difference, what is shared among both are the notions that time is branching to allow for reasoning about different hypothetical futures, and that a situation is a first-class object that can be quantified over. The treatment of situations as terms greatly increases the expressiveness of the formalization and distinguishes it from propositional treatments of dynamical systems [96].

While McCarthy considered a situation to be a complete state of affairs, he acknowledged that a situation calculus theory was not intended to supply a complete description of situations. Rather, he intended that a theory give a partial description of a situation, and would do so by stating facts about situations. This led to the notion of a fluent, named after Newton's use of the term to describe a physical quantity that depends on time. McCarthy described a fluent as "a predicate or function whose argument is a situation," giving examples of propositional fluents such as $raining(s)$ and $at(I, home, s)$ and functional fluents such as $time(s) = 1963.7205$. Propositional fluents (also known as relational fluents) are either true or false in a situation.

Although the papers by McCarthy, and McCarthy and Hayes, (and much later, Reiter [145,146]) all treated such fluents as predicates with a situation term as their final argument, many subsequent uses of the situation calculus proposed to reify relational fluents as first-order objects. McCarthy himself did so in later work, introducing a special binary predicate $Holds(s, p)$ to express the truth value of relational fluent p in situation s [104,118]. The advantage of reification is that it treats relational fluents as first-order objects that can be quantified over. This in turns allows the formalism to talk about properties of fluents, as is done in Lin's treatment of causation [95,96]. As discussed in Section 3, Moore [129], in his unification of the situation calculus and the possible-worlds account of knowledge, identifies situations as possible worlds, and quantifies over fluents in order to state the properties on accessibility relations that correspond to axioms of knowledge. Similarly, certain types of frame axioms are most easily stated by quantifying over fluents. It is sometimes difficult to tell from the presentation syntax whether or not relational fluents are reified because $Holds(s, p)$ is often replaced by the shorthand $p(s)$.

McCarthy originally treated actions as propositional fluents. However, in their 1969 paper, he and Hayes treated actions as terms. They introduced the functional fluent $result(p, \sigma, s)$, where the value of $result(p, \sigma, s)$ (later referred to as $Result(\sigma, s)$) is the situation that results when p carries out action σ starting in situation s.

2.2. Problems for theories of action: The frame, qualification, and ramification problems

The study of detailed examples in the situation calculus has led to the discovery of several important problems for action theories. Three of the best known problems — the frame problem, the qualification problem, and the ramification problem — were first extensively discussed in the context of the situation calculus, although they show up as well in other formalisms, such as the event calculus, Temporal Action Logics, and the fluent calculus. McCarthy was the first to discuss both the frame and qualification problems; the ramification problem was identified by Finger [36]. These problems, and McCarthy's approach to solving them, are discussed in this section; other researchers' approaches, often in languages other than the situation calculus, are discussed both in Section 2.3 and in Section 4.

2.2.1. The frame problem

An action theory will typically contain axioms specifying how actions change the state of the world. For example, putting one block on top of another will change the location of the first block. It will also change an aspect of the second block; it will no longer be a clear block.

$$Holds\big(s, IsClear(b2)\big) \quad\rightarrow\quad Holds\big(Result\big(puton(b1, b2), s\big), On(b1, b2)\big) \tag{1}$$

$$Holds\big(s, IsClear(b2)\big) \quad\rightarrow\quad \neg Holds\big(Result\big(puton(b1, b2), s\big), IsClear(b2)\big) \tag{2}$$

However, the vast majority of the features about the block and about the world will remain the same. This action will not change the color of the blocks, or their weight, or the location of the table, or the name of the U.S. president or the coloring of a Monarch butterfly.

The difficulty is that an action theory that merely gives effect axioms, along with a partial characterization of an initial situation, does not allow inferring anything about all the properties that do not change as a result of actions that are performed. For example, suppose that one has the axioms above, along with the following facts about some particular blocks:

$$Holds\big(S1, Color(BlockA, Red)\big) \tag{3}$$

$$Holds\big(S1, Color(BlockB, Blue)\big) \tag{4}$$

$$Holds\big(S1, IsClear(BlockA)\big) \tag{5}$$

$$Holds\big(S1, IsClear(BlockB)\big) \tag{6}$$

The theory given in (1) through (6) entails, as expected, that

$$Holds\big(Result\big(puton\big(BlockA, BlockB\big), S1\big), On(BlockA, BlockB)\big) \tag{7}$$

However, the theory does not entail that

$$Holds\big(Result\big(puton(BlockA, BlockB), S1\big), Color(BlockA, Red)\big) \tag{8}$$

or that

$$Holds\big(Result\big(puton\big(BlockA, BlockB\big), S1\big), Color(BlockB, Blue)\big) \tag{9}$$

In order to infer (8) or (9), one would have to add something to this theory, such as an axiom that states that the *puton* action does not change the color of blocks. For example, if one adds to the theory the axioms:

$$Holds\big(s, Color(b1, c)\big) \rightarrow Holds\big(Result\big(puton(b1, b2), s\big), Color(b1, c)\big) \tag{10}$$

$$Holds\big(s, Color(b2, c)\big) \rightarrow Holds\big(Result\big(puton(b1, b2), s\big), Color(b2, c)\big) \tag{11}$$

then one can infer that both the top and bottom blocks retain their color after a *puton* action.

For a somewhat richer theory that included other properties, such as the weight of blocks, or other actions, such as walking, one would have to add further axioms, specifying that the *puton* action did not change the weight of a block, and that the *walk* action changes neither the weight nor color of a block. There are potentially a very large number of such axioms for any action theory: around *nm* axioms, where *n* is the number of actions, and *m* is the number of fluents, in the domain.

McCarthy and Hayes [121] noted the representational and reasoning difficulties that would arise from having so many axioms, and called this the *frame problem*. The term *frame* referred to the approach that they suggested for solving the problem: having a frame or vector-like structure for each action, which specified the fluents that changed for each action, "all others being presumed unchanged," in McCarthy and Hayes's terms. From a back-formation of the term *frame problem*, axioms such as (10) and (11), which state that certain properties remain the same after specific actions have been performed, have become known as *frame axioms*.

"All others being presumed unchanged" is a loaded phrase, since one cannot presume the lack of change within standard logic. Formalizing this concept was to lead McCarthy to the development of circumscription, a nonmonotonic logic, and to a large body of work on the challenges of reasoning about action within nonmonotonic logics. (See Section 4 for more discussion on this topic.)

2.2.2. The qualification problem

A subset of axioms of a theory in the situation calculus specify the preconditions that must hold in order for the action to be successful. For example, the *puton* action of the blocks world requires that the blocks involved in the action be clear. The *qualification problem*, first introduced by McCarthy in his 1986 paper "Applications of Circumscription to Common Sense Reasoning" [108], is the problem of adequately representing the conditions for the successful performance of actions. McCarthy observed that in order to fully represent such conditions, an impractical and implausible number of qualifications would have to be included in the sentences expressing them. An example of the difficulty is the "potato in the tailpipe" scenario. In this scenario one gets into one's car, and turns the key. One typically expects the car to start, despite not knowing whether the battery is still charged, whether the ignition system is intact, or any of a potentially infinite set of *qualifications*, including the absence of a potato in the tailpipe. How can one represent and reasons about actions without elaborating the entire set of qualifications?

The essence of the solution that McCarthy proposed is to bundle all unexpected preconditions, such as potatoes in tailpipes or dead batteries, into a category of *abnormal* conditions, and to explicitly specify that such abnormal conditions do not hold as one of the antecedents of each effect axiom. The effect axiom for starting one's car, then, could be:

$$\neg Holds\big(s, ab(car)\big) \rightarrow Holds\big(Result\big(turnignition(car), s\big), Running(car)\big)$$

It should be noted that this is not merely a problem in reasoning about action. It holds as well in specifying inheritance hierarchies with exceptions. McCarthy gives the example of a taxonomy of birds. Typically birds fly, but of course a bird does not fly if it is a penguin or an ostrich or an emu. In fact, in order to conclude that a given bird flies, one must know not only that it is not a penguin or emu or ostrich or rhea or cassowary, but that it does not have a broken wing, is not a baby bird, is alive, and so on. As was the case with the axiom for starting one's car, McCarthy suggested that one use an abnormality predicate to formalize this rule, yielding[4]:

$$\neg ab(bird) \rightarrow Flies(bird)$$

The difficulty, then, is determining that the antecedents of these axioms — namely, that there is nothing abnormal about the car, and that there is nothing abnormal about the bird — hold. McCarthy provided a solution [108] that used his theory of circumscription, one of the nonmonotonic logics discussed in the next section.

2.2.3. The ramification problem

The ramification problem, the last of the classic trio of problems that are central to representing and reasoning about theories of action, is concerned with representing and reasoning about indirect consequences of actions. It was first discovered by Finger [36] in the context of abductive planning; Ginsberg and Smith [47] recognized that it presented a conceptual challenge for reasoning in the situation calculus.

The ramification problem arises in theories that have *state constraints*, axioms that relate two or more fluents within one state. For example [47], if a room vent is blocked, the room is stuffy. A theory may have many state constraints. The essence of the ramification problem is how to interpret the indirect effects of actions, manifested by the state constraints, in a manner that captures the intended interpretation and that minimizes change.

2.3. The development of the situation calculus and other action languages

Since McCarthy's introduction of the situation calculus, there has been significant work on logical formalizations of action. Most notable among these are the *event calculus* first proposed by Kowalski and Sergot [66], further extended by Shanahan and Miller [127,128,158], and recently explored by Mueller [133]; Sandewall's work on *Features and Fluents* [151,152] and the extensive subsequent work of Doherty and his colleagues on *Temporal Action Logics* (TAL) [25]; the $\mathcal{A}$ family of languages originally proposed by Gelfond and Lifschitz and extended by them, their colleagues and collaborators [42]; Hölldobler and Schneeberger's initial contributions [60] and Thielscher's subsequent work on the *fluent*

[4] McCarthy actually suggested an abnormality predicate that ranged over different *aspects*, representing the ways in which birds could be abnormal, such as species or age; the example given here is simplified.

calculus [169]; and perhaps most notably in the context of McCarthy's work, Reiter's axiomatization of the situation calculus[5] [146].

In general, these languages pointedly differ from the situation calculus in at least one respect; these differences may make such languages more suitable than the situation calculus for a particular representational choice or reasoning task. It is generally thought that action formalisms with branching time naturally support reasoning about hypotheticals and planning deductively, whereas a linear time structure more naturally supports reasoning about concurrent actions, narratives and observations, and planning via abduction, though these are not hard and fast rules.

2.3.1. Reiter's axiomatization of the situation calculus

McCarthy and Hayes's original situation calculus was proposed as a way to logically specify dynamical systems; but as Reiter notes [146], was predominantly viewed by the AI community as a theoretical tool without much practical importance. Reiter's account of the situation calculus axiomatized the language in ways that provided for reasonably efficient implementation through translations into highly compact Prolog code. Reiter's axiomatization of the situation calculus has several distinguishing features. First, situations are conceived as action histories. In contrast to McCarthy and Hayes's view, the world is conceived as originating in a distinguished situation, S_0. However, similarly to McCarthy and Hayes, Reiter's situtation calculus also subscribes to a branching time view of the world. Reiter axiomatizes this in terms of a set of foundational axioms that describes a tree of situations originating from S_0. The distinguished function $do(a, s)$ (the analog of McCarthy and Hayes's *Result*) maps an action and a situation into a successor situation.

A second distinguishing feature of Reiter's axiomatization of the situation calculus is his monotonic solution to both the frame problem and aspects of the qualification problem. The solution to the frame problem is embodied in a set of successor state axioms, one for each fluent, which state (intuitively) that a fluent F is true in the situation resulting from performing action a in situation s if and only if, a is among a specified set of actions that make F true, or F was already true and a is not among a specified set of actions that make F false. This solution to the frame problem [145] was derived from previous work by Pednault [135], Davis [13], and Haas [53], as elaborated by Schubert [157]. The success of the solution is predicated on the assumption that few things change as the result of performing an action.

Reiter addressed aspects of the qualification problem by appealing to Pednault and specifying action precondition axioms, one for each action, that state all and only the conditions under which an action is executable in a situation. Reiter's solutions to the frame and qualification problems necessitate $m + n$ axioms, where m is the number of fluents, and n is the number of actions, a substantial improvement over the nm axioms that concerned McCarthy and Hayes. Various solutions to the ramification problem have been proposed over the years. Lin and Reiter proposed a circumscriptive solution in [97], and Lin subsequently proposed a solution by appealing to an explicit representation of causality within the situation calculus [95]. Pinto proposed several solutions to the ramification problem for syntactically restricted state constraints [137]. McIlraith proposed compiling indirect effects into the successor state axioms of fluents, justifying the correctness of the compilation by appealing to prioritized circumscription and an extralogical specification of causal dependency [124]. Finally Ternovskaia provided independent justification of this approach via inductive definitions [167].

Reiter and various collaborators subsequently incorporated additional features into the language such as time, concurrency, and probabilities, to support specification of diverse dynamical systems. It has also been applied to a diversity of tasks including temporal projection, planning, and the characterization of database updates. The agent programming language Golog, developed by Reiter, Levesque, Lespérance, Lin, and Scherl [86], and its variants such as ConGolog, developed by de Giacomo, Lespérance, and Levesque [45], support the specification of complex actions in an ALGOL-like language. These complex actions echo some of the ideas underlying *strategies* as introduced by McCarthy and Hayes. In Golog, a complex action is represented as a macro that expands into a situation calculus formula that places constraints on the situation tree corresponding to the execution of the complex action. In ConGolog, complex actions are treated as terms in the language and their semantics is described using a transition semantics. Golog and its many variants have become the cornerstone of a significant amount of research on high-level robot control for Cognitive Robotics [83]. Golog has also been exploited extensively for web service composition [125].

2.3.2. The event calculus

As with the situation calculus, there are many variants of the event calculus, including the original version by Kowalski and Sergot [66], a simplified version, and a circumscriptive version [128,158]. A recent construction by Mueller also exists [133].

The event calculus is often described as a narrative-based formalism: in contrast to the situation calculus, its ontology of time is a single time line, rather than a branching temporal structure. McCarthy's *Result* function (similarly Reiter's *do* function) maps a situation and an action into a successor situation lying on a branch segment that corresponds to the way

[5] We henceforth refer to the axiomatization of the situation calculus described in [146] as "Reiter's axiomatization of the situation calculus." However as Reiter himself acknowledges, a number of the ideas surrounding the axiomatization of the situation calculus, the conceptualization of the basic action theory, and other subsequent extensions were developed by Reiter together with his many colleagues, collaborators and students. He particularly acknowledges Hector Levesque's significant intellectual contributions, Fiora Pirri for her collaboration in the development of the foundational axioms and important proofs deriving from these axioms [139], and previously Javier Pinto [137] and Fangzhen Lin [97] for their contributions to the development of foundational axioms for the situation calculus.

the world *might* be if particular actions were performed. However, there is no inherent distinction between what actually occurs and what hypothetically occurs. In contrast, in the simplified event calculus the predicate $Happens(a,t)$ asserts that action a definitely occurred at time t.

The timeline structure of the event calculus makes it particularly well suited to describing narratives, concurrent events, and triggered events, all of which are arguably somewhat more awkward to express in the situation calculus. It is also more suitable for characterizing planning as an abductive reasoning task [159], whereas planning is viewed as a deductive reasoning task in the situation calculus.

Fluents are reified in the event calculus. In the simplified event calculus, the predicate $HoldsAt(f,t)$ asserts that a fluent f holds at a given time point, t. Actions are represented as terms. The distinguished predicates $Initiates(a,f,t)$ and $Terminates(a,f,t)$ denote that a fluent f initiates and ceases to hold, respectively, when action a is executed at time t. While solutions to the frame problem in the event calculus differ depending on the formulation, the simplified event calculus solves the frame problem in a similar way to the axiomatic solution to the frame problem in Reiter's version of the situation calculus. In particular, a fluent f is true at (*HoldsAt*) time t if and only if it has been made true in the past and has not been made false in the interim. The distinguished predicate $Clipped(t_1, f, t_2)$ asserts that an event occurs that terminates f between t_1 and t_2. An extensive discussion of the nuances of the frame problem in the variants of event calculus and situation calculus can be found in [158].

The original event calculus was formulated as a set of Horn clauses that could be encoded as a logic program with negation as failure. The encoding of other variants of the event calculus in Prolog has also been realized, under suitable conditions. As with the situation calculus, the event calculus has been extended in various ways to represent continuous actions, concurrent actions, and triggered events. It has also been used to formally characterize and address a diversity of problems including database updates, as well as various planning and cognitive robotic tasks [128].

2.3.3. Temporal Action Logics

Temporal Action Logics, a class of logics for reasoning about action and change, evolved from Sandewall's work on *Features and Fluents* [151]. Sandewall proposed a number of logics, each defined semantically in terms of preferential entailment. The soundness and completeness of each logic was proven with respect to a set of intended conclusions. The logics originated from a narrative-based logical framework in which agent behavior was specified in terms of *action scenarios*.

Like the event calculus, Sandewall's logics and Temporal Action Logics are narrative in style, state based rather than action-history based, and have a linear time line. Fluents are reified; their truth or falsity is established using a $Holds$ predicate. By virtue of their linear time structure, Temporal Action Logics are typically well-suited to reasoning about narratives and observations, and for representing and reasoning about concurrent actions and delayed effects of actions. TAL's development has provided the semantic basis for TALPlanner [74] and for extensive practical work on planning and execution monitoring of autonomous unmanned aerial vehicles [27].

A full appreciation of TAL relies on familiarity with concepts of nonmonotonic reasoning such as circumscription, pointwise circumscription, and preferential entailment; see Section 4 for more detail. Preferential entailment, the methodology that underlies the development of the logics in TAL, characterizes a subset of preferred classical models. Each logic of TAL differs with respect to the preference relation to be minimized. PMON (Pointwise Minimization of Occlusion with Nochange premises) was one of the first logics proposed by Sandewall. Doherty subsequently developed an equivalent syntactic characterization of PMON in classical second-order logic by using a circumscription axiom to formalize preferential entailment. He showed that this was equivalent to a first-order pointwise circumscription axiom, enabling the use of standard first-order theorem proving to reason about PMON action narratives [24,28]. This characterization of PMON led to TAL. TAL and its many variants differ with respect to what is being minimized.

While solutions to the frame problem such as Reiter's are often portrayed in terms of a first-order axiomatization, they are often justified by appealing to circumscription (e.g., [97]) or some similar minimization policy. The TAL family of languages characterizes different minimization policies to address the frame, ramification, and qualification problems with respect to different expressiveness considerations, such as concurrency and causal constraints. Under certain conditions, the resultant, typically second-order, characterization can be shown to be first-order definable, or definable in a closed form, as Doherty did with PMON.

Axiomatizations of Temporal Action Logics typically exploit persistence statements that specify inertia and default value assumptions, and occlusion statements that characterize when a feature is exempt from this persistence. Doherty and colleagues have established a variety of conditions under which the TAL family of languages are first-order definable — pragmatic ways in which the elegance of circumscription can be preserved and utilized to characterize general solutions to problems in reasoning about action and change, and other topics in commonsense reasoning. A complete description of Temporal Action Logics can be found in [26].

2.3.4. Fluent calculus

The fluent calculus [168,169], another first-order formalism for reasoning about action and change, takes much of its basic ontology from the situation calculus, adding to this the concept of a *state* – intuitively a collection of fluents akin to McCarthy's original notion of a situation. As in the situation calculus, time is branching. In addition to the initial situation S_0 of Reiter's version of the situation calculus, the fluent calculus adds the notion of the empty state $\emptyset$, a constructor $\circ$, and a function *State*(s) to connect states with situations. Fluents are reified in the fluent calculus in order to have the explicit notion of a state.

Although the fluent calculus borrows much from Reiter's axiomatization of the situation calculus, an important difference between this modern-day situation calculus and the fluent calculus — and to a large measure, the motivation for the development of the fluent calculus — concerns their implementations. The update rules of the fluent calculus lead naturally to an implementation where its states are *progressed*; that is, one reasons about the facts that hold true in the state resulting from the performance of an action. In contrast, Reiter's axiomatization leads naturally to an implementation where situations are *regressed*; that is, one can rewrite a situation in terms of the preceding situation and the action that was performed in that situation. Under appropriate conditions, repeated regression in Reiter's axiomatization of the situation calculus eventually results in a formula in terms of the initial situation, S_0. Experiments have shown that in cases where agents execute long sequences of actions, tasks such as temporal projection are more quickly achieved via the fluent calculus' progression approach.

While the time structure of the fluent calculus is branching, the provision of state in fluent calculus supports the adoption of a linear time structure while preserving the solution to the frame problem. This supports narrative reasoning and abductive planning [171].

The fluent calculus derives its solution to the frame problem from the work of Bibel [6] and of Hölldobler and Schneeberger [60]. As with Reiter's situation calculus, the solution is addressed in terms of a set of axioms — state update axioms, rather than successor state axioms. However, whereas Reiter has one successor state axiom for each *fluent*, the fluent calculus has one state update axiom for each *action*. The existing solution to the qualification problem appeals to Default Logic [144]. There are several proposals for solutions to the ramification problem, most appealing to predicate completion and some form of second-order minimization, rather than circumscription.

A logic programming language FLUX has been developed [170], holding much the same relationship to the fluent calculus as Golog holds to the situation calculus. It has also been used for high-level programming of robots and other agents, but has particularly distinguished itself in its application to game-playing competition [156].

2.3.5. The $\mathcal{A}$ family of action languages

An important class of propositional action languages, developed by Gelfond, Lifschitz, and their colleagues and collaborators, is the $\mathcal{A}$ action language and its descendants [42,43,48]. The creators of this family of languages conceptualize action languages as formal models of those aspects of natural language that are used to talk about the effects of actions. Action theories are viewed as transition systems. Each transition system is represented as a finite directed graph whose vertices correspond to possible states of the system; the edges of the graph are interpreted in the same way as in the situation calculus tree — actions are terms and situations are action histories. The semantics of these languages are automata theoretic. Most of the languages are branching time. Since the languages are propositional, the frame problem is solved by simple set-theoretic operations. There are various approaches to the qualification and ramification problems, such as those using causal logic [101] or logic programming [5].

Gelfond and Lifschitz differentiate two kinds of action languages: *action description languages* serve to describe a transition system; and *action query languages* serve to state assertions about a transition system. Over the course of many years, Gelfond, Lifschitz and their collaborators have developed a variety of action description and query languages, and demonstrated their versatility with respect to a broad range of applications.

Languages in this family include $\mathcal{A}$, which augments the expressive power of STRIPS [35] with conditional actions in the spirit of Pednault's ADL [135]; $\mathcal{B}$, which supports the specification of indirect effects and which differentiates between *static* and *dynamic* causal laws; $\mathcal{C}$, which supports specification of nondeterministic and concurrent actions as well as allowing inertia to be postulated rather than assumed by default; and $\mathcal{C}+$, whose semantics is give by a reduction to causal theories [174].

One of the appeals of this family of languages is its simple formulation and the ease with which domains can be axiomatized. Further since these languages are not logic specific, they allow for translation to multiple formalisms, including Answer Set solvers and logic programs, and for easy establishment of proofs of correctness. For example, the highly optimized Causal Calculator (CCalc), which implements a fragment of the causal logic described in [48], often uses $\mathcal{C}+$ or its predecessor $\mathcal{C}$ as an input specification language [174].

3. Reasoning about knowledge

Since the 1960s, McCarthy has explored the relationship between intelligent agents and knowledge. His contributions have consisted of identifying new problems in reasoning about knowledge; suggesting solutions to these problems; and perhaps most important, formulating new representational schemes for knowledge and belief that have resulted in formal systems that have the expressive power of modal or higher-order logics while still remaining first order.

The problems that McCarthy identified include determining epistemic feasibility, reasoning about ignorance, reasoning about consistency of beliefs, and representing awareness and self-consciousness. Many of these themes first appeared in his and Patrick Hayes's groundbreaking paper "Some Philosophical Problems from the Standpoint of Artificial Intelligence." These themes were continued, and new problems introduced, in "Epistemological Problems of Artificial Intelligence," "Ascribing Mental Qualities to Machines," "First Order Theories of Individual Concepts and Propositions," and "Formalization of Two

Puzzles Involving Knowledge" [121,104,106,116]. As discussed below, many of these themes trace their origins to "Programs with Common Sense."

3.1. Representing an agent's knowledge

In order to reason about any problem involving knowledge, one must first be able to represent that knowledge. McCarthy's approach to representing the knowledge of an agent was shaped by the ongoing developments in the philosophical community at that time regarding modal logics and logics of knowledge. He suggested using Hintikka's approach to formalizing knowledge [59], in which "Know" is represented as a *modal operator* that ranges over sentences, and is given a *possible-worlds semantics*. Logicians had developed modal logics in order to investigate concepts of necessity and possibility [87], of obligation and permission [176,177], and of endurance and eventuality [141]; and had suggested several different axiom systems to characterize these operators. Informal attempts to give meaning, or semantics, to these operators have existed as long as modal logics have been investigated: discussions can be found in works by Diodorus Cronus, Jean Buridan, and Leibniz [140,61,78,64]. For example, Leibniz defined something as necessarily true if it was true in all possible worlds, possibly true if it was true in at least one possible world. Kripke [69–71] demonstrated how one could give a formal semantics to these modal operators by considering *accessibility relations* on possible worlds. Specifically, he defined something as necessarily true relative to a specific world W0 if it was true in all worlds accessible to W0. He then showed how different axioms systems on modal logic correspond to different restrictions on these accessibility relations. For example, the modal axiom $\mathit{Necessarily}\ (p) \rightarrow p$ (veridicality) corresponds to the property of reflexivity; that is, each possible world being accessible from itself; while the modal axiom $\mathit{Necessarily}(p) \rightarrow \mathit{Necessarily}(\mathit{Necessarily}(p))$ corresponds to the accessibility relation being transitive.

Hintikka [59] developed the first modal logic of knowledge and applied Kripke's ideas to his new logic. He considered an *epistemically accessible* relation on possible worlds, where a possible world W1 is knowledge accessible in W0 if for all one knows in W0, one might just as well be in W1; and defined knowledge in terms of that. Under his interpretation, a sentence $\mathit{Know}(a, p)$ is true in a possible world if it is true in all possible worlds epistemically accessible from that world. Kripke's results on the equivalence of various axiom systems to suitable restrictions on the accessibility relation carry over to Hintikka's logic.

In "Some Philosophical Problems from the Standpoint of Artificial Intelligence," McCarthy and Hayes suggested using Hintikka's logic for representing an agent's knowledge. Yet McCarthy was not entirely enthusiastic about all aspects of Hintikka's approach. In general, he has maintained an ambivalence toward any modal logic. His ultimate goal has been to formalize intelligent reasoning within first-order logic, or something as close to first-order logic as possible. In "Formalization of Two Puzzles Involving Knowledge" [116],[6] McCarthy showed how one could reason directly *within* possible worlds, using first-order logic, instead of reasoning in the modal logic itself. Glimpses of this approach can already be seen in "Some Philosophical Problems." In addition, he has advocated for *reification*, a method of treating complex logical structures as objects in a first-order language, so that they can be used as predicate arguments within the first-order logic [106]. For example, one can map sentences onto objects; this makes it possible to use predicates or functions, rather than modal operators, to express modality.

Both approaches have proved greatly influential in the development of formal AI. Moore [129] used McCarthy's technique of reasoning directly within possible worlds in his extensive solution to the knowledge preconditions problem, discussed below. Moore tightly integrated his solution to McCarthy's situation calculus by identifying possible worlds and situations. This permits one to reason about an agent's knowledge at a particular situation. Moore's approach has been used extensively by many at the University of Toronto, including Reiter, Levesque, and their collaborators [146,155,77], as well as by Davis [15]. While Moore, Reiter, and Levesque mostly express facts within the modal logic itself, Davis has developed techniques for expressing nearly all statements as sentences that are true within possible worlds, avoiding even the use of modal operators.

The use of possible worlds as models in which to check theorems of knowledge and belief, in contrast to the time-consuming operation of proving theorems within a modal logic, has been further advocated by Halpern and Vardi [56].

McCarthy's second technique for avoiding the use of modal logic has been used by several AI researchers. Both Konolige [65] and Morgenstern [131] have eschewed modal logic in favor of a syntactic logic in which *Know* is represented as a predicate that ranges over quoted sentences of first-order logic. Since the quotation mechanism can lead to paradox, techniques to ensure consistency — due to Tarski [166] and Kripke [73] — are used.

3.2. Knowledge preconditions

A central concern of McCarthy's early work on reasoning about actions and plans, present in "Programs with Common Sense," was the notion of *feasibility*. In order to reason that an agent can ultimately achieve a goal through a plan, one must be able to reason that the agent can perform the actions in the plan, that is, that the actions are feasible for the

[6] McCarthy started writing this paper in 1971; a reference in his paper "Modality, Si! Modal Logic, No!" [112] gives the date of this paper as 1978, although it does not appear to have been published before the 1990s.

agent. In the early problems that McCarthy considered, such as the getting-to-the-airport problem, and the monkey and bananas problem (in which a monkey plans to get an out-of-reach bunch of bananas by moving a box directly underneath the bananas, and climbing onto the box), McCarthy focused on the physical aspect of feasibility. For example, one might say that it is feasible to climb onto a box if one is at the same location as the box.

In addition to reasoning about physical feasibility, however, one must also reason about *epistemic feasibility*, whether or not an agent *knows* enough to perform a particular action. Consider the action of calling the White House switchboard. In order to perform this action, it is not sufficient that one be near a phone and that the phone be working (the physical preconditions), one must also know that the number of the White House switchboard is 202-456-1414. This is a *knowledge precondition* for the phone-calling action. McCarthy and Hayes discussed the knowledge preconditions problem,[7] the problem of how to represent and reason about the knowledge that is needed to perform actions, in "Some Philosophical Problems from the Standpoint of Artificial Intelligence" [121], focusing in part on introducing functions that are referentially transparent.

Moore [129] pioneered an extensive solution to this problem, suggesting that an agent knows how to perform an action if he knows a *rigid designator* [72] — a term that denotes the same entity in all possible worlds — for the action. In practice, this generally reduces to knowing rigid designators for the parameters of an action function. For example, knowing a rigid designator for the action of calling the White House switchboard reduces to knowing the specific numbers that must be pushed on the phone. Moreover, Moore posited that an agent knows how to do a multi-step plan if he knows enough to do each step at the time it comes up in the plan.

Morgenstern [131] extended the problem to consider multi-agent and communicative plans. Davis and Morgenstern [17] have characterized certain circumstances in which an agent knows enough to successfully execute a plan in which actions are delegated. Levesque and several of his students have studied various aspects of the knowledge preconditions problem. Specifically, Scherl and Levesque [155] have integrated Moore's theory with Reiter's solution to the frame problem within the situation calculus, demonstrating that Reiter's regression results [146] hold as well for knowledge-producing actions. Ghaderi, Levesque, and Lespérance have examined how agents' knowledge can affect their ability to execute a joint plan [44]. Levesque and Lakemeyer have studied the interaction between physical sensing and knowledge [82,77].

3.3. Reasoning about ignorance

An intelligent agent can reason not only from what he knows and what he knows others know, but from his ignorance and his knowledge of others' ignorance. Consider an agent A who wishes to get someone's telephone number. A is aware that P knows what the number is, but that Q does not. A rational plan would involve asking P for the information; in most cases, on the other hand, it would be irrational to ask Q.

McCarthy has been particularly interested in formalizing the reasoning required to solve a variety of knowledge puzzles that involve reasoning about knowledge and ignorance. In "Formalization of Two Puzzles Involving Knowledge," he discussed possible formalizations for two puzzles — *The Three Wise Men* and *Mr. S and Mr. P* — in which intelligent agents need to reason about the ignorance of others.

McCarthy's variant of the Three Wise Men puzzle runs as follows: A king arranges three wise men in a circle so that they can all see each other. He tells them that he is placing a black or white dot on each forehead, and that at least one dot will be white. Actually, all three dots are white. He then repeatedly asks them if they know the color of their dots. What do they say, and when?

The answer is that the first two times the king asks the question, the wise men all say they do not know the color of their dots; while the third time, each wise man says that he realizes that his dot is white. Why is this so? Each wise man Wi reasons as follows.

If there were only one white dot, the problem would be trivial. One of the wise men would see two black dots, and would immediately know that the color of his dot must be white. So he would immediately answer when the king asked him the color of his dot.

If there are two white dots, then there are two wise men who look around and see one black dot and one white dot. Each does not know the color of his dot, so what he does not know is whether there is one white dot or two white dots. But if there were only one white dot, then by the argument above, this would be immediately known. Therefore, if none of the wise men can answer the king's initial question, there must be two white dots. Both wise men with white dots realize this, so the second time the question is asked, they both say that they have white dots.

In the case given in the puzzle, each Wi knows that there are either two white dots or three white dots, depending on whether Wi himself has a black or white dot. But it has been established that if there are two white dots, the two wise men who have them will be able to answer the king's second question. Therefore, when no one is able to answer the king's second question, each wise man realizes that there must be three white dots. So, they are all able to answer the king's third question.

[7] The problem was so named by [129].

Indeed, the Three Wise Men puzzle and its variants, such as the Muddy Children problem, have been central to a line of research in formal AI that studies the circumstances in which agents can achieve common knowledge, and investigates applications in areas such as distributed systems and cryptography [32].

McCarthy argued that reasoning about ignorance is not only essential to the solution of brainteaser puzzles, but also for much commonsense reasoning. For example, if you are asked whether you know, at this moment, whether President Obama is standing up or sitting down, then, in the absence of unusual circumstances (you watch him over the internet or TV, or you are in the same room as him), you ought to realize that you do not know, and that no amount of thinking or reasoning it out is going to help you.

One can approach this problem using standard monotonic reasoning. Such a solution could involve stating some principle from which one can infer ignorance. For example, Davis [12] has developed a theory of knowledge and perception which supports this reasoning. In his theory, agents can reason about their ignorance based upon their knowledge of the limits of their perceptual power. Using a dual structure of physical layouts and situations, which places layouts within a temporal and epistemic structure, one can formulate an axiom that states that if physical situations are possible, and do not contradict perceptions, then they cannot be known to be false. This explains why you cannot know whether Obama is sitting or standing.

Reasoning about one's ignorance is also closely connected to nonmonotonic reasoning, which is used to reason about what is typically true. (Nonmonotonic reasoning is discussed in more detail in the following section.) While most nonmonotonic theories, such as circumscription [107], default logic [144], and NML [123] appeal to the consistency of a set of formulas, Moore's formulation of nonmonotonic reasoning, autoepistemic logic [130], appeals to the consistency of an agent's knowledge. It explicitly considers how an agent can reason defeasibly from his ignorance. One of Moore's motivating examples was the reasonableness of a statement like "If I had an older brother, I would know about it," and of the reasoning process that would let you determine from first, that statement; second, an axiomatization of knowledge, such as weak S5; and third, the fact that you have no knowledge of an older brother, the conclusion that you do not have an older brother. The relationship between ignorance, or more precisely, all that an agent knows, and nonmonotonic reasoning was further explored by Levesque [85]. Both Perlis [136] and Lifschitz [91] have also explored the connection between McCarthy's circumscription and autoepistemic logic.

The question of determining whether Obama is sitting or standing has a multiple-agent analogue as well. Not only do you not know whether Obama is standing, you also are quite sure that Putin, if he does not see Obama, does not know the answer either. How do agents make conclusions about other agents' ignorance? Several researchers have examined this question, and the related general question of how you can know whether an agent does not know some statement, within the context of multiple agent nonmonotonic reasoning, including [132,68,75,54].

3.4. Awareness and contexts

McCarthy has repeatedly argued [121,105,111,119,120] that intelligent agents need a considerable degree of self-awareness — to determine whether one's belief set is consistent; to decide on a problem-solving strategy (for example, to decide that one can postpone coloring a part of the map-coloring problem (like states that have only three bordering states, such as California) [113]); to understand, as in the Three Wise Men problem, that other agents are using the same reasoning processes that oneself is, but differ only, possibly on the color of the dots that they see; to realize that no amount of thinking things out will help one know whether Obama is sitting or standing.

McCarthy has suggested using a structure called contexts to enable modeling self awareness [109,110]. In ordinary discourse, sentences are true or false not absolutely, but relative to a particular context. For example, specific facts about a person's weight may be true in the context of earth, but not in the context of the moon. It is true that Sherlock Holmes lived at 221B Baker Street in the context of Conan Doyle's detective stories, but not in the context of the real world, where Sherlock Holmes never existed.

A sentence in the language of contexts is of the form $ist(c,p)$, where c is a context, such as the world that Conan Doyle described, and p stands for a proposition. Various techniques related to reification can ensure that such sentences remain first-order. One can reason with propositions within a context, or move into different contexts.

Inspired by McCarthy, Lifschitz [89] and Shoham [164] did early work in the theory of contexts. This research was continued by two of McCarthy's students, Guha [51], who incorporated his work on contexts into the foundations of Cyc [80], a comprehensive knowledge base containing million of facts about the everyday world, and Buvac [8,7].

Other researchers have sought to formalize specific types of awareness. In two award-winning papers, Levesque [84] and Fagin and Halpern [31] both used the notion of awareness in order to deal with the consequential closure problem: the anomalies that result when an agent's knowledge is closed under logical consequence. These approaches formalize the observation that agents are typically consciously aware of only a portion of the facts that they know, and tend to reason with these facts.

The explicit reasoning about whether one's beliefs are consistent, and the reasoning needed to maintain consistency as an agent acquires new and possibly conflicting pieces of information, is the concern of the field of belief revision [1,62], which is a subfield of nonmonotonic reasoning.

4. Nonmonotonic reasoning

4.1. The need for nonmonotonic reasoning

At the time of the initial presentation of "Programs with Common Sense," at least one researcher recognized that in one important respect, commonsense reasoning was different than mathematical reasoning. This difference lay in the need for exceptions to express commonsense knowledge. Mathematical truths are absolute; not so, commonsense facts.

Bar-Hillel, one of the original attendees at the symposium at which McCarthy presented his paper, argued [4]:

> It sounds rather incredible that the machine could have arrived at its conclusion — which, in plain English, is "Walk from your desk to your car!" — by sound deduction! This conclusion surely could not possibly follow from the premise in any serious sense. Might it not be occasionally cheaper to call a taxi and have it take you over to the airport? Couldn't you decide to cancel your flight or to do a hundred other things?

Bar-Hillel was alluding to the many exceptions that could exist in any realistically complex situation. AI researchers realized the truth of Bar-Hillel's argument as they worked through initial simple examples. (See Sandewall's description, in this volume, of the Stanford group's recognition of the need for nonmonotonic reasoning in the early 1970s.)

This recognition led McCarthy and several other researchers, most notably Reiter [144] and McDermott and Doyle [123], to develop several different types of *nonmonotonic* reasoning systems. In classical systems of formal logic, a set of conclusions is monotonic in the set of assumptions — that is, the more premises one has, the more one can conclude. In contrast, nonmonotonic reasoning systems have the property that more premises can result in *fewer* conclusions. For example, in a nonmonotonic reasoning system, one could infer, or jump to the conclusion, that if birds typically can fly, unless they are penguins, and Tweety is a bird, then Tweety can fly. But if an extra premise is added, that Tweety is a penguin, one would withdraw the conclusion that Tweety can fly.

The particular type of nonmonotonic reasoning that McCarthy developed is known as circumscription and is discussed below.

4.2. Circumscription

4.2.1. The intuition behind circumscription

There is a close connection between reasoning about what is typically true and reasoning about exceptions to a rule. For example, to express the fact that birds typically fly, one can say that any bird flies unless it is exceptional, e.g.,

$$Bird(x) \land \neg ab(x) \to Flies(x)$$

(The predicate *ab* indicates that the bird is *abnormal* or *exceptional* in some way.) One could then add facts specifying when a bird is abnormal:

$$Penguin(x) \to ab(x)$$

$$Cassowary(x) \to ab(x)$$

$$Injured(x) \to ab(x)$$

Suppose that one is given these rules, along with a fact about a specific bird Tweety:

$$Bird(Tweety)$$

Can one now conclude that Tweety flies? The answer is no: these statements together do not entail that $Flies(Tweety)$. There is nothing in the theory that precludes Tweety being a penguin or cassowary or injured bird; indeed, there is nothing in the theory that precludes $ab(Tweety)$. And unless one knows that $\neg ab(Tweety)$, one cannot conclude that $Flies(Tweety)$.

Intuitively, one expects $\neg ab(Tweety)$ to be true because there is no known fact indicating that Tweety is abnormal. But that is a commonsense intuition; it is not justified by the rules of first-order logic.

McCarthy sought to extend first-order logic into a formal system that did entail this commonsense intuition. His approach, formalized in his system of nonmonotonic reasoning, which is known as circumscription, rests upon the following insights[8]:

1. It is often necessary to determine that an object is not abnormal (exceptional) in some way.
2. In order to do so, one needs to ensure that the extension of the abnormality predicate is as small as possible, that is, to *circumscribe* the extension of the predicate as much as one can.
3. One can do so by adding formulas to a set of first-order sentences so that the resulting theory entails that a predicate has the smallest extension possible.

[8] The following discussion relates to predicate circumscription; these can be suitably modified for domain and formula circumscription.

The theory that formalizes these insights is McCarthy's theory of circumscription.

4.2.2. Basics of circumscription

McCarthy has proposed three different types of circumscription, domain circumscription [104], predicate circumscription [107] and formula circumscription [108].

The most commonly used formulation, predicate circumscription, is concerned with limiting the extension of a predicate as much as possible. For example, if one has a theory with a single sentence $P(a)$, where P is a predicate and a is a constant, then circumscribing the predicate P should limit the extension of P to consist of exactly a.

One circumscribes a predicate P by adding to the theory a formula that states that any predicate p that holds for the objects for which P holds must be at least as large as P. Thus, assuming that $A(P)$ is a sentence containing the predicate P, the circumscription of P in $A(P)$ is the sentence

$$A(P) \wedge \neg\exists p\big[A(p) \wedge p < P\big]$$

Note that this sentence contains quantification over predicates and is therefore second-order.

There is a natural model-theoretic interpretation of circumscription. Consider a language L with predicate P and constants a, b, and c, representing pairwise distinct elements of the universe and consider a theory T of L consisting of the single formula $P(a)$. Then there are models satisfying T in which $P(a)$ holds alone; in which $P(a)$ and $P(b)$ hold; in which $P(a)$ and $P(c)$ hold; and in which $P(a)$, $P(b)$, and $P(c)$ all hold. One can think of the circumscription of P as inducing an ordering on the models: the smaller the extension of P, the more preferred the model. In particular, the model of T in which only $P(a)$ holds is the minimal (most preferred) element in this ordering. McCarthy [107], drawing partly from Martin Davis [19], gives a formal definition of the model-theoretic characterization of circumscription.

Particularly because it is often difficult to compute circumscription (see the discussion in Section 4.2.4), it is often more convenient to work directly with the model-theoretic definition of circumscription. (The general strategy of checking the truth of a model instead of theorem proving using axioms and rules of deduction is also advocated by Halpern and Vardi (1991)).

4.2.3. McCarthy's original formulations of circumscription

The presentation of the definition of circumscription given in the preceding paragraphs is due to Lifschitz [93] and is a simplified version of the second-order version of circumscription given by McCarthy in 1986 [108].

McCarthy, in his first major paper on circumscription, "Circumscription: A Form of Non-monotonic Reasoning" [107], had defined the circumscription of P in $A(P)$ using axiom schemas rather than a second-order axiom. Assume a predicate symbol $P(x1,\dots,xn)$, written as $P(\bar{x})$, and let $A(\Phi)$ be the result of replacing all occurrences of P in A by the predicate expression Φ.

Then the circumscription of P in $A(P)$ is the sentence schema

$$A(\Phi) \wedge \forall\bar{x}\big(\Phi(\bar{x}) \to P(\bar{x})\big) \to \forall\bar{x}\big(P(\bar{x}) \to \Phi(\bar{x})\big)$$

In his 1986 paper "Applications of Circumscription to Formalizing Common Sense Knowledge," McCarthy modified and generalized this definition in several ways. For the new definition, $A(P)$ is a formula of second-order logic, P is a tuple of some of the free predicate symbols in $A(P)$, $E(P,x)$ is a wff in which P and a tuple x of variables are free.

Then, the circumscription of $E(P,x)$ relative to $A(P)$ is defined as:

$$A(P) \wedge \forall P'\big[A\big(P'\big) \wedge \big(\forall x E\big(P',x\big) \to E(P,x)\big) \to \big(\forall x E\big(P',x\big) = E(P,x)\big)\big]$$

This formulation differs from the earlier formulation in two ways. First, the definition is given using a second-order formula rather than an axiom schema. Second, the definition allows for the circumscription of formulas rather than just predicates. Predicate circumscription is a special case of formula circumscription when $E(P,x)$ has the form $P(x)$.

McCarthy also introduced two other important extensions of circumscription. First, he allowed the specification of a circumscriptive policy in which the extension of some predicates may vary and the extension of other predicates is fixed. This is essential to getting circumscription to work as intended. For example, consider the Tweety example introduced earlier in this section. If one minimizes the extension of *ab*, the extension of *Flies* will get larger: if a particular bird is not abnormal in some way, it will fly.

Second, McCarthy introduced the concept of prioritized circumscription. There are times when there is a conflict between minimizing the extensions of two different predicates. For example [93], consider a case where one considers two information sources to be reliable unless they are abnormal in some way:

$$\mathit{Reported}1(x) \wedge \neg \mathit{ab}1(x) \to \mathit{Happened}(x)$$

$$\mathit{Reported}2(x) \wedge \neg \mathit{ab}2(x) \to \mathit{Happened}(x)$$

Now suppose that these information sources broadcast different stories. That is,

$$Reported1(A)$$

$$\neg Reported1(B)$$

$$Reported2(B)$$

$$\neg Reported2(A)$$

Suppose further that the reports are contradictory:

$$\neg\big(Happened(A) \wedge Happened(B)\big)$$

In some models, those in which the extension of $ab1$ is minimal, $Happened(A)$ will be true; in some models, those in which the extension of $ab2$ is minimal, $Happened(B)$ will be true.

For circumstances in which this is not desirable — where e.g., one wishes to say that the first source is more reliable than the second, and that therefore, the extension of $ab1$ ought to be "more minimal" than the extension of $ab2$, one can use prioritized circumscription.

4.2.4. Using circumscription and its extensions

McCarthy's initial work on circumscription has inspired much research on variations and extensions to the original concept of circumscription; methods for computing circumscription; the relationship of circumscription to other nonmonotonic formalisms; and the use of circumscription in formalizing commonsense domains, particularly in theories of action.

A large portion of this research is due to Vladimir Lifschitz, who transformed circumscription from an interesting technical idea into a usable technique, in two important ways. First, he realized that circumscription, in McCarthy's formulations, could not always be used to get the results that were intuitively desired. To remedy this problem, he introduced variations or extensions of circumscription, such as pointwise circumscription [90], which offers a more fine-grained circumscriptive policy than ordinary circumscription. It allows circumscription on a tuple-by-tuple basis, rather than of the entire predicate. Indeed, pointwise circumscription is quite powerful for expressing sophisticated circumscriptive policies: Lifschitz used it in the first solution he proposed to the *multiple extension* problem, discussed below. Second, he provided methods for computing circumscription.

Computing circumscription becomes a concern when one considers anything beyond the simplest of toy problems. The examples that appear in McCarthy's 1980 paper on circumscription are meant to be illustrative and are relatively straightforward. In such cases, figuring out what one obtains by adding circumscription — that is, determining what is entailed by the circumscribed theory — is relatively simple. However, in the 1986 paper, in which McCarthy was already considering more practical and complex problems, such as using circumscription to formalize inheritance hierarchies with exceptions, and to solve the frame problem, it had already become considerably harder to compute what a given circumscriptive policy actually entailed. As a rule, the more complex the circumscriptive policy, and the more complex the theory in which one performs the circumscription, the more difficult it is to compute circumscription.

To enable the computation of circumscription, Lifschitz [88] developed methods to allow replacement of a second-order circumscriptive formula with an equivalent first-order formula, and to rewrite complex circumscriptive policies in simpler terms. Lifschitz's results, however, were somewhat limited in that they covered only isolated categories of circumscriptive formulas. Doherty, Lukaszewicz, and Szalas [29] have developed more general reduction methods and provide an algorithm which takes any second-order circumscriptive formula and either provides a first-order reduction or terminates in failure. They have also studied reduction methods for domain circumscription [25].

4.3. Applying circumscription: Challenges and solutions

4.3.1. Circumscription and unintended multiple extensions

McCarthy's development of circumscription was largely motivated by his desire to provide simple and efficient methods for dealing with the qualification and frame problems. As discussed above, McCarthy suggested using and circumscribing an abnormality predicate to handle the qualification problem. Likewise, to handle the frame problem, he suggested using and circumscribing another type of abnormality predicate. He suggested formalizing the principle of inertia, namely, that most fluents remain the same when actions are performed, in the following way: One would write down, for each fluent, an axiom that stated that the fluent remained the same when an action was performed, unless the action was abnormal in some way with respect to a particular fluent. One could then furnish axioms specifying the actions that were abnormal.

For example, a formalization of the blocks world could contain axioms saying that an action does not change an object's location unless it is abnormal in some respect, and that the *Move* action is abnormal in this respect; that an action does not change an object's color unless it is abnormal in some respect, and that painting an object is abnormal in this respect.

This works for some simple examples, as McCarthy showed in his paper "Applications of Circumscription to Common-Sense Reasoning." However, it quickly became evident that it was easy to come up with many examples — some quite simple — for which this approach would not work at all.[9] For these examples, the expected conclusion did not hold.

[9] By the time [108] was published, articles citing the difficulties with McCarthy's approach were being accepted for publication as well. McCarthy's presentation of his ideas for handling the frame problem had predated the publication of his paper by several years.

Circumscription — and nonmonotonic reasoning in general — did not seem to yield the set of expected models. That is, the models of the circumscribed theory included unexpected models. This became known as the *multiple extension* problem.

The multiple extension was independently discovered by Lifschitz [90], who discovered it in the context of using qualification axioms for a problem in the blocks-world domain; and Hanks and McDermott [57], who discovered it in the context of axiomatizing the principle of inertia, for the well-known Yale shooting problem. In both cases, considering a sequence of two actions is enough to cause problems for the nonmonotonic reasoning.

Lifschitz considers a very simple blocks-world domain where the causal theory contains the following axioms:

$$\neg ab\big(Move\big(x, Top(y)\big), s\big) \rightarrow Holds\big(Result\big(Move\big(x, Top(y)\big), s\big), On(x, y)\big)$$

That is, unless the action of moving a block x to the top of another block y is abnormal, as a result of this action, x will be on top of y. (Note that this is a qualification axiom.)

$$Holds\big(s, On(y, x)\big) \rightarrow \neg Holds\big(s, Clear(x)\big)$$

That is, if one block is on top of a second block, the second block is not clear, and

$$\neg Holds\big(s, Clear(x)\big) \rightarrow ab\big(Move(x, loc), s\big)$$

That is, if a block is not clear, moving the block will be abnormal (i.e., the *Move* action will not work as expected; in particular, in this case, it will not effect any changes).

Consider what happens given the following description of an initial situation:

$$Holds\big(S0, Clear(A)\big)$$
$$Holds\big(S0, Clear(B)\big)$$

Consider what is true now about $S1$ and $S2$, where

$$S1 = Result\big(Move\big(A, Top(B)\big), S0\big)$$
$$S2 = Result\big(Move\big(B, Top(A)\big), S1\big)$$

One would expect that since *ab* will be minimized, the first *Move* action will not be abnormal, and will result in A being on top of B; and that therefore, the second *Move* action *will* be abnormal (i.e., will not result in B being on top of A).

In fact, this is not entailed by circumscription. The circumscribed theory gives two classes of models: the expected one in which $ab(Move(B, Top(A)), S1)$ holds; that is, it is the second *Move* action that does not go successfully; and the unexpected one in which $ab(Move(A, Top(B)), S0)$ holds; that is, it is the first *Move* action that does not go successfully. In this unexpected model, since the first *Move* action doesn't work, both blocks stay clear; thus, the second *Move* action is successful!

The well-known Yale shooting problem uses a version of an abnormality predicate to formalize the principle of inertia:

$$Holds(s, f) \wedge \neg ab(f, e, s) \rightarrow Holds\big(Result(e, s), f\big)$$

One also has causal rules saying that loading a gun causes the gun to be loaded and that shooting a loaded gun causes a turkey to die:

$$Holds\big(Result(Load, s), Loaded\big)$$
$$Holds(s, Loaded) \rightarrow \neg Holds\big(Result(Shoot, s), Alive\big) \wedge ab(Alive, Shoot, s)$$

Finally, there is a boundary condition stating that the turkey is originally alive in $S0$:

$$Holds(S0, Alive)$$

One then considers whether or not

$$Holds\big(Result\big(Shoot, Result\big(Wait, Result(Load, S0)\big)\big), Alive\big)$$

One would expect that the above would be false. *Loaded* is true once the *Load* action has taken place, and ought to persist until the *Shoot* takes place, entailing that the turkey is no longer alive. As with Lifschitz's blocks world problem, however, there are two models, the intended model in which the *Shoot* action is abnormal with respect to the turkey being alive, and the unintended model, in which the *Wait* action is abnormal with respect to the gun being loaded, and in which, therefore, the turkey does not die.

The existence of unintended multiple extensions presents a serious problem for proponents of nonmonotonic reasoning. If straightforward axiomatizations of even simple problems do not yield the intended conclusions, the goal of using first-order logic to encode commonsense knowledge would seem to be much more difficult to achieve.

4.3.2. Careful representation for expected conclusions

Despite the initial brouhaha surrounding the Yale shooting problem and related scenarios, and the apprehension, in some circles (e.g., [122]), that formal logic was not after all a suitable vehicle for commonsense reasoning, dealing with the problem has not been as difficult as was originally feared. The first insight that researchers had, concerning the specific problems, was that unintended interpretations arose when abnormalities were posited in an unexpected order. For example, in the Yale shooting problem, it seems more natural to consider the model in which the gun remains loaded as long as possible [162,90]. But these initial solutions were not quite satisfactory, either. What turned out to be crucial was the formalization of the notion of causation. If one has a sufficiently strong representation of causation in one's theory, one can prefer models in which changes are caused, rather than simply holding in a particular model. Shanahan [158] provides an excellent account of how one can use nonmonotonic reasoning to solve the frame problem without falling into Yale-shooting-type problems. The realization that the concept of causation has an essential place in any theory of temporal reasoning now seems obvious. It is sometimes hard today to explain to one's students, or to people new to the field of knowledge representation, why the Yale shooting problem was ever considered such a challenging problem. Today, the existence of the Yale shooting problem is perhaps most instructive as an example of the importance of doing knowledge representation right, and in particular, of developing an axiomatization only after one has carefully thought through one's model of the world.

4.4. Other nonmonotonic formalisms

Although circumscription has been the nonmonotonic formalism that is most often associated with McCarthy, a brief discussion of other major nonmonotonic systems is appropriate in this article for several reasons. First, it can be legitimately argued that all nonmonotonic formalisms stem from McCarthy's early work. His "Programs with Common Sense" made clear, as Bar-Hillel pointed out, that commonsense reasoning required the ability to represent and reason with exceptions; researchers in the Stanford AI Laboratory were to come to this conclusion themselves during the next decade. "Some Philosophical Problems," moreover, made explicit the need for such a formalism. Second, an important component of the research that McCarthy has inspired includes papers that compare circumscription to other nonmonotonic formalisms. Third, other systems used for implementing nonmonotonic reasoning — in particular, answer-set programming systems — can be used for reasonably efficient applications that do commonsense reasoning, thus enabling progress toward realizing McCarthy's vision of the advice taker.

Default logic [144] introduces the concept of a nonmonotonic inference rule, known as a default rule. The default rule consists of an assumption, an appeal to the consistency of some formula, and a conclusion. For example, the rule *Birds typically fly* can be written as

$$\frac{Bird(x) : Flies(x)}{Flies(x)}$$

A default theory consists of a set of sentences in a first-order logic and a set of default rules. The set of extensions of a default theory is given in terms of a fixed-point definition.

Autoepistemic logic [130] characterizes nonmonotonicity using the belief operator L of a doxastic or epistemic logic.[10] The focus is on *stable sets* of sentences [165], those sentences that can be viewed as the beliefs of a rational agent, and the *stable expansions* of a premise set. Properties of stable sets include consistency and a version of negative introspection, the principle that if an agent does not believe something, that fact of non-belief is itself believed by the agent. (See Section 3.)

To formalize the Tweety example, one can represent the rule that birds typically fly as

$$L\big(Bird(x)\big) \wedge \neg L\big(\neg Flies(x)\big) \rightarrow Flies(x)$$

That is, if I believe that x is a bird and I don't (have a reason to) believe that x cannot fly, then (I can conclude that) x flies.

If one starts out with the premise set consisting of this rule and the premise

$$Bird(Tweeety)$$

any stable expansion will contain the conclusion

$$Flies(Tweety)$$

What characterizes all forms of nonmonotonic reasoning is, of course, the nonmonotonicity of the entailment or consequence operator between premises and conclusion. The nature of this consequence operator, and its properties, has been studied by [163] and [67]. Alchourrón, Gärdenfors, and Makinson [1] and Katsuno and Satoh [63] have also studied the connection between consequence relations and belief revision.

It has turned out to be particularly fortuitous for McCarthy's research program, and his focus on nonmonotonic reasoning, that it is easy to add a nonmonotonic feature — negation as failure — into Prolog systems. The relation between

[10] Autoepistemic logic was inspired by the work of McDermott and Doyle [123], who first suggested using modal logic to capture nonmonotonicity. They interpreted the modal operator as a consistency operator and did not have an analogue to Moore's development of a stable expansion.

nonmonotonic reasoning and logic programming was first noted in the late 1970s by Ray Reiter [143] and Keith Clark [10], at about the time that McCarthy was first developing circumscription. There has been active research on the relationship between these two fields as evidenced by the biennial conference started in 1991, the International Conference on Logic Programming and Nonmonotonic Reasoning, that has been devoted to the topic.

Of particular importance is the large body of work on answer-set programming. It has been recognized since the late 1970s that the negation-as-failure feature of logic programs could be used to implement nonmonotonicity. Starting in the late 1980s and early 1990s, Gelfond and Lifschitz [40,41] demonstrated the connections between the class of logic programs known as answer-set programs, and nonmonotonic logics with fixed-point semantics, in particular, autoepistemic logic and default logic.

5. Related research interests

In addition to his main areas of research, McCarthy has explored several related research topics, briefly discussed below.

5.1. Domain axiomatizations and elaboration tolerance

From his early experiences in attempting to formalize non-trivial problems within first-order logic, McCarthy realized the importance of constructing axiomatizations of actual domains such as physical reasoning and spatial reasoning. McCarthy recognized that such axiomatizations are necessary, that they are hard, and that the exercise of producing an axiomatization often leads to the discovery of important research problems. As the field of formal knowledge representation advanced, and researchers often focused on studying properties of specific formalisms, McCarthy encouraged researchers to focus on developing domain axiomatizations. To this end, he founded in 1991 the biennial symposium on Logical Formalizations of Commonsense Reasoning (known as the Commonsense Symposium).

One of the primary difficulties in developing domain axiomatizations is ensuring that they are not brittle. That is, it should be relatively easy to add facts without having to alter the foundations of the axiomatization. McCarthy [117,114] called theories that satisfied this property *elaboration tolerant*, which he characterized as follows [114]: “A formalism is elaboration tolerant to the extent that it is convenient to modify a set of facts expressed in the formalism to take into account new phenomena or changed circumstances.”

As his motivating example, McCarthy considered the Missionaries and Cannibals problem. The problem, as formulated by Amarel [2], is of three missionaries and three cannibals who need to cross a river in a rowboat, subject to the constraints that only two people fit in a rowboat and that cannibals eat missionaries if the missionaries are ever outnumbered by the cannibals on either bank of the river. Amarel had originally developed a representation of triples representing the number of missionaries, cannibals, and boats on the starting bank of the river. Allowable actions were represented as transformations on the triples. Solving the problem, then, was an exercise in heuristic search. McCarthy, in his first paper on circumscription [107], had argued that this representation abstracted away the most interesting parts of the problem, and that a theory of circumscription could enable reasoning with a richer representation. For example, it should not be impossible to state that there is a bridge linking the two banks of the river one mile away — which would make the standard solution to the problem unnecessary, since all missionaries and cannibals could simply walk over the bridge without any fuss. However, proper usage of circumscription should appropriately minimize any unmentioned bridges, leaving the problem solver again with just the rowboat and the constraints.

In his work on elaboration tolerance, McCarthy returned to the consideration of the bridge as well as various other elaborations. What if there is one oar on each bank of the river? The missionaries must send the cannibal to get the oar. What if there are four missionaries and four cannibals? The problem becomes unsolvable. What if the missionaries and cannibals all wear hats and exchange hats at some point during the journey? This has no relevance to the solution: how does an intelligent agent know this?

McCarthy’s goal was to create a domain axiomatization that enabled solutions to all elaborations by making only relatively minor changes to the theory. His challenge has been taken up by a number of researchers, including Lifschitz [94], who showed how most of the elaborations could be solved using a straightforward representation in the input language of the Causal Calculator [100,48], and Gustafsson and Kvarnström [52], who showed that an object-oriented representation could be used to efficiently represent and reason with the elaborations. In addition, Ernie Davis has developed a methodology that includes considering a wide range of *variants* or elaborations from the initial statement of the problem through its ultimate solution. (See [14] and his problems on the Commonsense Problem Page [126].)

5.2. Learning from experience

McCarthy’s later research has renewed its focus on developing general-purpose intelligent robots. In “The Well-designed Child” [120], McCarthy considers how one might build a robot that can learn from experience. Such a robot would not be a Lockean *tabula rasa*, as was conceived in the 1950s, but instead endowed with innate structures with which the robot could make sense of the world. Examples of such innate structures and functionalities would be the ability to perceive motion as continuous, the notion of persistence, and a grammar of goal regression, which allows agents to plan to perform an action by recursively planning to satisfy the actions’ preconditions.

6. Papers in this special issue

This special issue comprises 16 technical papers, each one with its origins firmly rooted in the logic-based approach to AI established by John McCarthy more than 50 years ago. In what follows we briefly discuss the contributions of each paper in relation to McCarthy's work.

6.1. Papers on representing and reasoning about action

The rich tradition of research in theories of action that stems from McCarthy's invention of the situation calculus is apparent in several papers in this issue. As noted in Section 2.3, McCarthy's original conception of the situation calculus and its elaboration by McCarthy and Hayes, provided the conceptual foundation for a number of different logic-based action formalisms. Over the years, these formalisms have in turn formed the basis for study of various issues relating to the topic of reasoning about action and change.

The contribution by Alfredo Gabaldon [38], "NonMarkovian Control in the Situation Calculus," investigates a modification of Reiter's axiomatization of the situation calculus and its consequences. Most theories of action satisfy the *Markov property*, i.e., the property that executability conditions and effects of actions are fully determined by the present state of the system. Gabaldon observes that it is often desirable to directly axiomatize actions whose effects and executability conditions violate this property, in that they may depend on past and even alternative hypothetical situations. To this end, he removes the Markov property restriction from the situation calculus and generalizes Reiter's regression operator, a critical computational device in Reiter's formalism, to deal with non-Markovian theories. He further identifies a syntactically restricted class of formulas for which this regression is applicable, and illustrates the utility of his work in terms of a number of small examples. Gabaldon also provides an implementation of a formula evaluator for this class of formulas. This work is significant because it defines the underlying principles of non-Markovian action theories, an important class of action theories for many practical applications.

While the frame problem has been extensively studied and is considered by most to be solved, the qualification problem still presents challenges. In their paper "Modular-$\mathcal{E}$ and the Role of Elaboration Tolerance in Solving the Qualification Problem," Antonis Kakas, Loizos Michael, and Rob Miller re-examine the qualification problem in the context of $\mathcal{M}$odular-$\mathcal{E}$ ($\mathcal{ME}$). $\mathcal{ME}$, which originates in the event calculus, is a specialized, model-theoretic logic for reasoning about actions that supports nondeterministic domains in which concurrency, static laws, indirect effects, and narrative can be expressed. Kakas et al.'s paper is particularly relevant to this special issue in honor of John McCarthy because it articulates the role of *elaboration tolerance*, a concept introduced by McCarthy [115], in addressing the qualification problem. As explained in Section 5, a formalism is elaboration tolerant to the extent that it is convenient to modify a set of facts expressed in the formalism to take into account new phenomena or changed circumstances. Kakas et al.'s $\mathcal{ME}$ language is designed to be modular and as such has a high degree of elaboration tolerance. Kakas et al. exploit this property in order to provide principled solutions to different aspects of the qualification problem.

A problem of practical import for reasoning about action is planning — the problem of generating a structured collection of actions, together with constraints on those actions, in order to achieve a specified goal. McCarthy's work on commonsense reasoning has often emphasized the importance of programs that can plan intelligently. Many of the recent advances in automated planning have focused on classical planning over propositional theories, where complete specification of the initial state is given. The contribution to this special issue by Phan Huy Tu, Tran Cao Son, Michael Gelfond, and A. Ricardo Morales, "Approximation of Action Theories and its Application to Conformant Planning" [173], examines the problem of conformant planning: generating a sequence of actions that is guaranteed to achieve a goal, given only a partial specification of the initial state of the world. Planning is often studied algorithmically; in contrast, what makes this work particularly interesting and relevant to this special issue is that the authors approach the problem from a knowledge representation perspective. In particular, Tu et al., whose work appeals to $\mathcal{AL}$, a propositional action language from the $\mathcal{A}$ family of action description languages, propose an approximation of their action theory, which they encode as a logic program under answer set semantics.

The diversity of action formalisms in use today, and as reflected in this collection, provides partial motivation for Michael Thielscher's contribution to this collection, "A Unifying Action Calculus" [172]. In his paper Thielscher proposes an action calculus that unifies well-defined classes of a number of action formalisms, including the situation calculus, the fluent calculus, and the event calculus. In doing so, Thielscher is able to establish formal correspondences between these formalisms and to facilitate their comparison and translation. This unifying calculus also provides an opportunity to address fundamental problems in reasoning about action and change once rather than within each of the individual formalisms.

6.2. Papers on reasoning about knowledge

The challenge of representing and reasoning about the interplay between an agent's knowledge and its actions are discussed in a number of McCarthy's writings. This special issue includes several papers that explore issues related to reasoning about knowledge explicitly or as a tool for simplifying the axiomatization of theories of action.

The paper by Gerhard Lakemeyer and Hector Levesque, "A Semantic Characterization of a Useful Fragment of the Situation Calculus with Knowledge," fits well into this second category. It presents a new logical variant of the situation calculus with knowledge, called $\mathcal{ES}$ [76]. This system is proved correct in the sense that it can be embedded in the axiomatization of the situation calculus proposed by Reiter. However, the semantics of $\mathcal{ES}$ differs from the standard Tarskian semantics of predicate formulas, which allows a formula to be interpreted in structures with different domains. Instead, $\mathcal{ES}$ uses standard names and a substitutional interpretation of first-order quantifiers. This feature of $\mathcal{ES}$ allows the authors to give simpler proofs of important metamathematical theorems, such as Reiter's regression theorem.

In the contribution to this special issue by Steven Shapiro, Maurice Pagnucco, Yves Lespérance, and Hector Levesque, "Iterated Belief Change in the Situation Calculus" [160], Shapiro et al. provide an account of belief change in the context of agents performing actions in the world. In doing so, they attempt to address some of the perceived shortcomings of previous approaches to belief change in the context of a theory of action. The authors build on Scherl and Levesque's extension to Reiter's axiomatization of the situation calculus [154,155], which deals with knowledge expansion. The authors augment this variant of the situation calculus with a notion of plausibility over situations. In doing so they are able to account for nested belief, belief introspection, mistaken belief, belief revision and belief update, together with iterated belief change.

Afsaneh Shirazi and Eyal Amir's contribution to this special issue, "First-Order Logical Filtering" addresses the problem of belief revision from a computational perspective [161]. The authors examine the problem of updating a set of possible world states following a sequence of executed actions and observations. While such *logical filtering* of first-order theories is intractable in the general case, the authors identify useful special cases where the task is tractable, and present polynomial-time algorithms for filtering first-order belief states in such cases. They further identify cases where their representations are compact; this has implications for propositional filtering as well. Their analysis is based on Reiter's axiomatization of the situation calculus.

Under the possible-worlds semantics of knowledge an agent knows all logical consequences of his knowledge, including all tautologies; that is, the agent is *logically omniscient*. This view of knowledge is often considered problematic both because it does not accord with our experience of how intelligent beings reason (e.g., no person knows all consequences of Peano's axioms) and because it disregards limitations on an agent's computational resources. The logical omniscience problem is discussed in the paper contributed to this issue by Joe Halpern and Riccardo Pucella, "Dealing With Logical Omniscience: Expressiveness and Pragmatics" [55]. They review several approaches to dealing with logical omniscience, argue that these approaches are not equi-expressive when probabilities are taken into consideration, and discuss the pragmatics of dealing with logical omniscience.

6.3. Papers on nonmonotonic reasoning

Definitions of nonmonotonic entailment come in two flavors. Translational definitions, such as program completion [10] and circumscription, use nonmonotonic syntactic transformations of logical formulas and show how to formalize nonmonotonic reasoning in classical logic. Fixed-point definitions, such as default logic [144] and autoepistemic logic [130], define models or extensions of a theory by fixed-point conditions. Answer sets have been long viewed as part of the fixed-point tradition in nonmonotonic reasoning.

The paper by Paolo Ferraris, Joohyung Lee, and Vladimir Lifschitz, "Stable Models and Circumscription" [34], presents answer sets in a different light. It turns out that the answer sets of a logic program can be described using a syntactic transformation that is very similar to the one employed originally by McCarthy in the definition of circumscription. The authors argue that the new circumscription-style approach to answer-set programming may have several advantages. These include giving a new and broader perspective on the place of stable models within the field of nonmonotonic logic; providing a unified framework for useful answer-set programming techniques in different languages, such as combining choice rules with aggregates; and applicability to non-Herbrand models.

In their contribution "From Answer Set Logic Programming to Circumscription via Logic of GK," Fangzhen Lin and Yi Zhou [98] show that the general stable model semantics can be embedded in Lin and Shoham's logic of Grounded Knowledge (GK). This embedding allows them to obtain a way to check, in classical propositional logic, whether any two programs are strongly equivalent. It provides for a mapping from general logic programs to propositional circumscription. This mapping, when extended to the first-order case yields a semantics to first-order general logic programs that is similar to one previously proposed by Ferraris, Lee, and Lifschitz in previous work that appeared in IJCAI-07 [33].

Answer-set programming became a successful programming methodology after the creation of efficient software for generating stable models. "The Semantics and Complexity of Aggregates in Answer Set Programming" is contributed by Wolfgang Faber, Gerald Pfeifer, and Nicola Leone [30], who have designed a successful and expressive software system that implements answer-set programming, DLV [81]. Their paper presents a method for representing and reasoning about and with *aggregates*, a collection of items that together form some quantity or total. Aggregates are useful for formalizing many applications: an example of a business rule containing an aggregate is the principle that one owns a controlling interest in a company if one owns at least 50 percent of the stock in the company and its subsidiaries. Most work on aggregates has focused on the non-recursive case; Faber et al. investigate the more difficult recursive case.

6.4. Papers focusing on domain axiomatizations

Two papers in this issue, by Ernest Davis [16] and by Pedro Cabalar and Paulo E. Santos [9], present domain axiomatizations of the kind discussed in Section 5.1.

Davis's "How Does a Box Work?", a study in qualitative physical reasoning, formalizes the knowledge necessary to reason about plans to transport items by placing them in an open box and carrying the box from one location to another. The paper demonstrates how difficult it can be to develop a robust theory that supports the seemingly simple inference that such plans will work, but is sufficiently elaboration tolerant so that it can support the right conclusions for plans that are executed in similar situations — for instance, when the box is turned upside down, or covered with a lid, or carried over rough terrain. The paper is a model account of the process of domain formalization, particularly in its exposition of a pre-formal "debugging" exercise, in which the author considers all the ways in which the proposed plan might go wrong, and decides to deal with each potential "bug" by specifying stricter conditions, making the plan more specific, or making the conclusions less certain, say by performing the inference in a default logic. An additional contribution is the development of an ontology of *heaps* of objects, along with axioms that support reasoning about the way these heaps behave. This allows reasoning about the plan to transport objects in a box in much the same commonsensical way that people do, without worrying about the exact position of each object in the box over time.

Cabalar and Santos's "Formalizing the Fisherman's Folly Puzzle" develops an axiomatization for temporal and spatial reasoning than can support solutions to several puzzles involving strings, holes, poles and various objects. The challenge of the Fisherman's Folly Puzzle is removing a ring from a wooden post with a hole through which runs a string that is bounded by two disks. In exploring this domain, Cabalar and Santos deal with the difficulties of axiomatizing both rigid objects (e.g., poles) and non-rigid objects (e.g., string). Their representation includes string-segment objects which can be created or annihilated over time. This allows one to represent the changing configuration of the string as it passes through disks and/or the disks are compressed.

6.5. Paper on combining logic with other methods

McCarthy's work on designing a robot that can learn from experience, described in Section 5, is the inspiration for Fiora Pirri's contribution to this issue, "The Well-designed Logical Robot" [138]. Pirri considers the problem of how agents can learn action theories from observations. For example, people can learn to open a door by observing others grasping a door handle, pressing down, and pushing (or pulling). Indeed, from these same observations, dogs can learn how to open a door, even though they cannot grasp and press down on a handle in the same way. The learning involves labeling constructs of importance (e.g., the door, the handle, the motion of grasping, the effect of pushing the door). Pirri presents a three-step framework for modeling this intelligent learning process, comprised of transforming a set of observed phenomena to a Hidden Markov Model (HMM) of the observed events, transforming the HMM to a parametric probability model, and transforming this probabilistic model to a first-order model consisting of sentences in the situation calculus which could presumably be used as input to a McCarthyian advice taker.

Pirri's work is notable both in its attempt to bridge the gap between the formal KR community and other parts of the AI community that are more focused on quantitative methods and applications, and in its attempt to lay out a practical road map for realizing John McCarthy's vision of the advice taker.

6.6. Papers giving historical perspectives

The special issue includes two papers that give historical perspectives on John McCarthy's work [153,175].

Erik Sandewall's "From Systems to Logic in the Early Development of Nonmonotonic Reasoning" provides an account of the prehistory of nonmonotonic reasoning and circumscription, illustrating through several examples how researchers came to realize that standard logic was not sufficiently powerful to model many sorts of intelligent reasoning. Nonmonotonic reasoning is often presented in a vacuum, as if it sprang full-blown from the foreheads of McCarthy [107], Reiter [144], and McDermott and Doyle [123], but of course this is not the case. Sandewall, whose own contributions to nonmonotonic reasoning go back to the earliest years of the field, and who was an active member of the Stanford AI Laboratory at the time that these ideas were just beginning to see light within the AI and knowledge representation communities, explains the context and the historical development of the ideas underlying circumscription, as well as several of the problems associated with reasoning about actions, including the frame problem and the qualification problem. This paper also gives a particularly interesting perspective on how the distinction between normal and non-normal default theories relates to the multiple extension problem.

From the early days of AI, there has been a strong synergy between the formal AI and the analytical philosophy communities. McCarthy has especially encouraged this synergy: "Some Philosophical Problems from the Standpoint of Artificial Intelligence" points out the relevance to formal AI of work in the philosophical community, such as Kripke's possible worlds semantics for modal logic, Hintikka's logics of knowledge and belief, Prior's tense logic, and Rescher's work on counterfactuals [59,70,71,142,147]. Indeed, the topics that McCarthy and Hayes laid out in their landmark paper have continued to occupy McCarthy for the last 40 years. Although McCarthy soon eschewed the formal construct of modal logic, choosing instead to use constructs, such as reification, that allowed him to stay within first-order logic, his work continues to be informed by related work in the philosophical community. Johan van Benthem's "McCarthy Variations in a Modal Key" is therefore of particular interest: it discusses several of McCarthy's best-known contributions, including the situation calculus and circumscription, from the perspective of modal logic, and traces the connection between the historical developments of McCarthy's formalisms, on the one hand, and modal logic, on the other.

7. Conclusion

The papers included in this special issue represent a tiny fraction of the many works that have been inspired by John McCarthy's body of research. There are doubtless many more papers to come that will owe their origins to McCarthy's ideas, not only arising from the sizable corpus of works that McCarthy has already inspired, but also from the many ideas in his papers that have yet to be investigated. His legacy, the field of formal knowledge representation, is indeed a rich one. We, members of the research community that he created, thank him for it.

Acknowledgements

We would like to thank the reviewers of the submissions to this special issue for their thorough reviews, and the authors of the articles that appear in this special issue for patiently waiting while the last of the papers were finalized. We thank Vladimir Lifschitz for his contributions to this article and for his assistance in the preparation of the special issue; Chitta Baral, Ernie Davis, Patrick Doherty, Jeff Finger, Pat Hayes, Gerhard Lakemeyer, and Michael Thielscher for their helpful comments; and Tony Cohn and Ray Perrault for supporting the creation of this volume. Finally, we thank John for providing the inspiration for a generation of AI researchers who have followed in the path that he first conceived and created.

References

[1] C.E. Alchourrón, P. Gärdenfors, D. Makinson, On the logic of theory change: Partial meet contraction and revision functions, Journal of Symbolic Logic 50 (2) (1985) 510–530.
[2] S. Amarel, On representation of problems of reasoning about action, in: D. Michie (Ed.), Machine Intelligence, vol. 3, Edinburgh University Press, Edinburgh, 1971, pp. 131–171.
[3] Aristotle, The Organon or Logical Treatises of Aristotle, Kessinger Publishing, 2007.
[4] Y. Bar-Hillel, J. McCarthy, O. Selfridge, Discussion of the paper: Programs with common sense, in: V. Lifschitz (Ed.), Formalizing Common Sense. Intellect, 1998, pp. 17–20.
[5] C. Baral, M. Gelfond, Logic programming and reasoning about actions, in: D. Gabbay, L. Vila (Eds.), Handbook of Temporal Reasoning in AI, Elsevier, 2005, pp. 389–428.
[6] W. Bibel, A deductive solution for plan generation, New Generation Computing 4 (1986) 115–132.
[7] S. Buvac, Resolving lexical ambiguity using a formal theory of context, in: K. van Deemter, S. Peters (Eds.), Semantic Ambiguity and Underspecification, Center for the Study of Language and Information, Stanford, 1996.
[8] S. Buvac, V. Buvac, I.A. Mason, Metamathematics of contexts, Fundamenta Informaticae 23 (2–4) (1995) 263–301.
[9] P. Cabalar, P.E. Santos, Formalising the fisherman's folly puzzle, Artificial Intelligence 175 (1) (2011) 346–377 (this issue).
[10] K. Clark, Negation as failure, in: H. Gallaire, J. Minker (Eds.), Logic and Data Bases, Plenum Press, New York, 1978, pp. 293–322.
[11] F. Copleston, Medieval Philosophy, Harper and Collins, 1961.
[12] E. Davis, Inferring ignorance from the locality of visual perception, in: Proceedings of the Seventh National Conference on Artificial Intelligence (AAAI-88), AAAI Press/MIT Press, 1988, pp. 786–790.
[13] E. Davis, Representations of Commonsense Knowledge, Morgan Kaufmann, San Francisco, 1990.
[14] E. Davis, The naive physics perplex, AI Magazine 19 (3) (1998) 51–79.
[15] E. Davis, Knowledge and communication: A first-order theory, Artificial Intelligence 166 (2005) 81–139.
[16] E. Davis, How does a box work? A study in the qualitative dynamics of solid objects, Artificial Intelligence 175 (1) (2011) 299–345 (this issue).
[17] E. Davis, L. Morgenstern, A first-order theory of communication and multi-agent plans, Journal of Logic and Computation 15 (5) (2005) 701–749.
[18] M. Davis, A computer program for Presburger's algorithm, in: Summaries of Talks Presented at the Summer Institute for Symbolic Logic, 1957, 2nd ed., Institute for Defense Analysis, 1960.
[19] M. Davis, The mathematics of non-monotonic reasoning, Artificial Intelligence 13 (1–2) (1980) 73–80.
[20] M. Davis, The early history of automated deduction, in: A. Robinson, A. Voronkov (Eds.), Handbook of Automated Reasoning, Elsevier, 2001.
[21] M. Davis, G. Logemann, D. Loveland, A machine program for theorem proving, Communications of the ACM 5 (7) (1962) 394–397.
[22] M. Davis, H. Putnam, Computational methods in the propositional calculus, Renssalaer Polytechnic Institute, Unpublished report, 1958.
[23] M. Davis, H. Putnam, A computing procedure for quantification theory, Journal of ACM 7 (3) (1960) 201–215.
[24] P. Doherty, Reasoning about action and change using occlusion, in: Proceedings of the Eleventh European Conference on Artificial Intelligence (ECAI-94), 1994, pp. 401–405.
[25] P. Doherty, J. Gustafsson, L. Karlsson, J. Kvarnström, Tal: Temporal action logics language specification and tutorial, Electronic Transactions on Artificial Intelligence 2 (1998) 273–306.
[26] P. Doherty, J. Kvarnström, Temporal action logics, in: V. Lifschitz, F. van Harmelen, F. Porter (Eds.), The Handbook of Knowledge Representation, Elsevier, 2008, pp. 709–757 (Ch. 18).
[27] P. Doherty, J. Kvarnström, F. Heintz, A temporal logic-based planning and execution monitoring framework for unmanned aircraft systems, Autonomous Agents and Multi-Agent Systems 19 (3) (2009) 332–377.
[28] P. Doherty, W. Lukaszewicz, Circumscribing features and fluents, in: Proceedings of the First International Conference on Temporal Logic (ICTL-94), vol. 827, 1994, pp. 82–100.
[29] P. Doherty, W. Lukaszewicz, A. Szalas, Computing circumscription revisited: A reduction algorithm, Journal of Automated Reasoning 18 (3) (1997) 297–336.
[30] W. Faber, N. Leone, G. Pfeifer, Semantics and complexity of recursive aggregates in answer set programming, Artificial Intelligence 175 (1) (2011) 278–298 (this issue).
[31] R. Fagin, J.Y. Halpern, Belief, awareness, and limited reasoning: Preliminary report, in: Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI-85), 1985, pp. 491–501.
[32] R. Fagin, J.Y. Halpern, Y. Moses, M.Y. Vardi, Reasoning About Knowledge, The MIT Press, Cambridge, Massachusetts, 1995.
[33] P. Ferraris, J. Lee, V. Lifschitz, A new perspective on stable models, in: Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07), 2007, pp. 372–379.
[34] P. Ferraris, J. Lee, V. Lifschitz, Stable models and circumscription, Artificial Intelligence 175 (1) (2011) 236–263 (this issue).
[35] R. Fikes, N.J. Nilsson, STRIPS: A new approach to the application of theorem proving to problem solving, Artificial Intelligence 2 (3–4) (1971) 189–208.

[36] J.J. Finger, Exploiting constraints in design synthesis, Ph.D. thesis, Stanford University, Stanford, CA, USA, 1987.
[37] G. Frege, Begriffsschrift, in: J.V. Heijenoort (Ed.), From Frege to Gödel: A Sourcebook in Mathematical Logic, 1879–1931, Harvard University Press, 1967, English title: Concept Script, a formal language of pure thought modelled upon that of arithmetic.
[38] A. Gabaldon, Nonmarkovian control in the situation calculus, Artificial Intelligence 175 (1) (2011) 25–48 (this issue).
[39] H. Gelernter, Realization of a geometry theorem proving machine, in: International Federation for Information Processing Congress (IFIP), 1959, pp. 273–281.
[40] M. Gelfond, V. Lifschitz, Compiling circumscriptive theories into logic programs, in: Proceedings of the Seventh National Conference on Artificial Intelligence, AAAI-1988, AAAI Press/MIT Press, 1988, pp. 455–459.
[41] M. Gelfond, V. Lifschitz, Classical negation in logic programs and disjunctive databases, New Generation Computing 9 (3/4) (1991) 365–386.
[42] M. Gelfond, V. Lifschitz, Representing action and change by logic programs, Journal of Logic Programming 17 (2–4) (1993) 301–321.
[43] M. Gelfond, V. Lifschitz, Action languages, Electronic Transactions of Artificial Intelligence 2 (1998) 193–210.
[44] H. Ghaderi, H.J. Levesque, Y. Lespérance, A logical theory of coordination and joint ability, in: Proceedings of the Twenty-Second Conference on Artificial Intelligence (AAAI-07), AAAI Press, 2007, pp. 421–426.
[45] G. de Giacomo, Y. Lespérance, H.J. Levesque, Congolog a concurrent programming language based on the situation calculus, Artificial Intelligence 121 (1–2) (2000) 109–169.
[46] P. Gilmore, A proof method for quantification theory: its justification and realization, IBM Journal of Research and Development 4 (1) (1960).
[47] M.L. Ginsberg, D.E. Smith, Reasoning about action I: A possible worlds approach, Artificial Intelligence 35 (2) (1988) 165–195.
[48] E. Giunchiglia, J. Lee, V. Lifschitz, N. McCain, H. Turner, Nonmonotonic causal theories, Artificial Intelligence 153 (1–2) (2004) 49–104.
[49] K. Gödel, The completeness of the axioms of the functional calculus of logic, in: J.V. Heijenoort (Ed.), From Frege to Gödel: A Sourcebook in Mathematical Logic, 1879–1931, Harvard University Press, 1967, pp. 582–591.
[50] K. Gödel, On formally undecidable propositions of Principia Mathematica and related systems, in: J.V. Heijenoort (Ed.), From Frege to Gödel: A Sourcebook in Mathematical Logic, 1879–1931, Harvard University Press, 1967, pp. 596–616.
[51] R.V. Guha, Contexts: A formalization and some applications, Ph.D. thesis, Stanford University, 1991.
[52] J. Gustafsson, J. Kvarnström, Elaboration tolerance through object-orientation, Artificial Intelligence 153 (1–2) (2004) 239–285.
[53] A.R. Haas, The case for domain-specific frame axioms, in: F. Brown (Ed.), The Frame Problem in Artificial Intelligence: Proceedings of the 1987 Workshop, Morgan Kaufmann, 1987, pp. 343–348.
[54] J.Y. Halpern, G. Lakemeyer, Multi-agent only knowing, Journal of Logic and Computation 11 (1) (2001) 41–70.
[55] J.Y. Halpern, R. Pucella, Dealing with logical omniscience: Expressiveness and pragmatics, Artificial Intelligence 175 (1) (2011) 220–235 (this issue).
[56] J.Y. Halpern, M.Y. Vardi, Model checking vs. theorem proving: A manifesto, in: Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning, 1991, pp. 325–334.
[57] S. Hanks, D.V. McDermott, Default reasoning, nonmonotonic logics, and the frame problem, in: Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86), 1986, pp. 328–333.
[58] P. Hayes, L. Morgenstern, On John McCarthy's 80th Birthday, in Honor of his Contributions, AI Magazine 28 (4) (2007) 95–102.
[59] J. Hintikka, Knowledge and Belief, Cornell University Press, 1962.
[60] S. Hölldobler, J. Schneeberger, A new deductive approach to planning, New Generation Computing 8 (3) (1990) 225–244.
[61] G.E. Hughes, The modal logic of Jean Buridan, in: Atti del Convegno internazionale di storia della logica: le teorie delle modalità, Bologna, 1989, pp. 93–111.
[62] H. Katsuno, A.O. Mendelzon, On the difference between updating a knowledge base and revising it, in: Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning (KR-91), 1991, pp. 387–394.
[63] H. Katsuno, K. Satoh, A unified view of consequence relation, belief revision and conditional logic, in: Proceedings of the Twelfth International Joint Conference on Artificial Intelligence (IJCAI-91), 1991, pp. 406–412.
[64] W. Kneale, M. Kneale, The Development of Logic, Oxford, 1962.
[65] K. Konolige, A deduction model of belief and its logics, Ph.D. thesis, Stanford University, 1984, also published as SRI Technical Report 326.
[66] R.A. Kowalski, M.J. Sergot, A logic-based calculus of events, New Generation Computing 4 (1) (1986) 67–95.
[67] S. Kraus, D.J. Lehmann, M. Magidor, Nonmonotonic reasoning, preferential models and cumulative logics, Artificial Intelligence 44 (1–2) (1990) 167–207.
[68] S. Kraus, D. Perlis, J.F. Horty, Reasoning about ignorance: A note on the Bush–Gorbachev problem, Fundamentae Informaticae 15 (3–4) (1991) 325–332.
[69] S. Kripke, A completeness theorem in modal logic, Journal of Symbolic Logic 24 (1) (1959) 1–14.
[70] S. Kripke, A semantical analysis of modal logic I: Normal modal propositional calculi, Zeitschrift fur Mathematische Logic und Grundlagen der Mathematik 9 (1963) 67–97.
[71] S. Kripke, Semantical considerations on modal logic, Acta Philosophica Fennica 16 (1963) 83–94.
[72] S. Kripke, Naming and Necessity, Harvard University Press, Cambridge, 1972.
[73] S. Kripke, Outline of a theory of truth, Journal of Philosophy 19 (1975) 690–716.
[74] J. Kvarnström, P. Doherty, Talplanner: A temporal logic based forward chaining planner, Annals of Mathematics and Artificial Intelligence 30 (1–4) (2000) 119–169.
[75] G. Lakemeyer, All they know: A study in multi-agent nonmonotonic reasoning, in: Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93), 1993, pp. 376–381.
[76] G. Lakemeyer, H. Levesque, A semantic characterization of a useful fragment of the situation calculus with knowledge, Artificial Intelligence 175 (1) (2011) 142–164 (this issue).
[77] G. Lakemeyer, H.J. Levesque, AOL: A logic of acting, sensing, knowing, and only knowing, in: Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning (KR-98), 1998, pp. 316–329.
[78] G.W. Leibniz, Discourse on Metaphysics, 1686, translated by George R. Montgomery, revised by Albert R. Chandler, 1924.
[79] G.W. Leibniz, New Essays on Human Understanding, 1705.
[80] D.B. Lenat, Cyc: A large-scale investment in knowledge infrastructure, Communications of the ACM 38 (11) (1995) 32–38.
[81] N. Leone, G. Pfeifer, W. Faber, T. Eiter, G. Gottlob, S. Perri, F. Scarcello, The DLV system for knowledge representation and reasoning, ACM Transactions on Computational Logic 7 (3) (2006) 499–562.
[82] H. Levesque, What is planning in the presence of sensing? in: Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI-96), AAAI Press, 1996, pp. 1139–1146.
[83] H. Levesque, G. Lakemeyer, Cognitive robotics, in: F. van Harmelen, V. Lifschitz, B. Porter (Eds.), Handbook of Knowledge Representation, Elsevier, 2008, pp. 869–886 (Ch. 23).
[84] H.J. Levesque, A logic of implicit and explicit belief, in: Proceedings of the Fourth National Conference on Artificial Intelligence (AAAI-84), 1984, pp. 198–202.
[85] H.J. Levesque, All I know: A study in autoepistemic logic, Artificial Intelligence 42 (2–3) (1990) 263–309.
[86] H.J. Levesque, R. Reiter, Y. Lespérance, F. Lin, R.B. Scherl, Golog: A logic programming language for dynamic domains, Journal of Logic Programming 31 (1–3) (1997) 59–83.

[87] C.I. Lewis, C.H. Langford, Symbolic Logic, Dover reprint, New York, 1932.
[88] V. Lifschitz, Computing circumscription, in: Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI-85), 1985, pp. 121–127.
[89] V. Lifschitz, On formalizing contexts, Manuscript, 1986.
[90] V. Lifschitz, Pointwise circumscription: Preliminary report, in: Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86), 1986, pp. 406–410, extended version in: M. Ginsberg (Ed.), Readings in Nonmonotonic Reasoning, Morgan Kaufmann, 1987, pp. 179–193.
[91] V. Lifschitz, Between circumscription and autoepistemic logic, in: Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning (KR-89), 1989, pp. 235–244.
[92] Vladimir Lifschitz (Ed.), Formalizing Common Sense: Papers by John McCarthy, Ablex, 1990.
[93] V. Lifschitz, Circumscription, in: D.M. Gabbay, C. Hogger, J. Robinson (Eds.), Nonmonotonic Reasoning and Uncertain Reasoning, in: Handbook of Logic in Artificial Intelligence and Logic Programming, vol. 3, Clarendon Press, Oxford, 1994, pp. 297–352.
[94] V. Lifschitz, Missionaries and cannibals in the causal calculator, in: Proceedings of the Seventh International Conference on Principles of Knowledge Representation and Reasoning (KR-00), 2000, pp. 85–96.
[95] F. Lin, Embracing causality in specifying the indirect effects of action, in: Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95), 1995, pp. 1985–1993.
[96] F. Lin, Situation calculus, in: V. Lifschitz, F. van Harmelen, F. Porter (Eds.), The Handbook of Knowledge Representation, Elsevier, 2008, pp. 649–669 (Ch. 16).
[97] F. Lin, R. Reiter, State constraints revisited, Journal of Logic and Computation 4 (5) (1994) 655–678.
[98] F. Lin, Y. Zhou, From answer set logic programming to circumscription via logic of GK, Artificial Intelligence 175 (1) (2011) 264–277 (this issue).
[99] D.W. Loveland, A simplified format for the model elimination theorem-proving procedure, Journal of the ACM 16 (3) (1969) 349–363.
[100] N. McCain, Causality in commonsense reasoning about actions, Ph.D. dissertation, University of Texas at Austin, 1997.
[101] N. McCain, H. Turner, A causal theory of ramifications and qualifications, in: Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95), 1995, pp. 1978–1984.
[102] J. McCarthy, Programs with common sense, in: Proceedings of the Teddington Conference on the Mechanization of Thought Processes, Her Majesty's Stationery Office, London, 1959, pp. 75–91.
[103] J. McCarthy, Situations, actions, and causal laws, 1963. Tech. rep., Stanford University, reprinted in: M. Minsky (Ed.), Semantic Information Processing, MIT Press, Cambridge, Mass., 1968, pp. 410–417.
[104] J. McCarthy, Epistemological problems of artificial intelligence, in: Proceedings of the Fifth International Joint Conference on Artificial Intelligence (IJCAI-77), 1977, pp. 1038–1044.
[105] J. McCarthy, Ascribing mental qualities to machines, in: M. Ringle (Ed.), Philosophical Perspectives in Artificial Intelligence, Humanities Press, 1979, pp. 161–195.
[106] J. McCarthy, First order theories of individual concepts and propositions, in: Machine Intelligence, vol. 9, Ellis Horwood, Chichester, 1979, pp. 129–148.
[107] J. McCarthy, Circumscription: A form of non-monotonic reasoning, Artificial Intelligence 13 (1–2) (1980) 23–79.
[108] J. McCarthy, Applications of circumscription to common sense reasoning, Artificial Intelligence 28 (1) (1986) 89–116.
[109] J. McCarthy, Generality in artificial intelligence, Communications of the ACM 30 (12) (1987), revised and expanded version of McCarthy's 1971 Turing Award lecture. Also printed in Vladimir LIfschitz (Ed.), Formalizing Common Sense.
[110] J. McCarthy, Notes on formalizing context, in: Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence (IJCAI-93), 1993, pp. 555–562.
[111] J. McCarthy, Making robots conscious of their mental states, Machine Intelligence 15 (1995) 3–17.
[112] J. McCarthy, Modality, si!, modal logic, no!, Studia Logica 59 (1) (1997) 29–32.
[113] J. McCarthy, Coloring maps and the Kowalski doctrine, in: V. Lifschitz (Ed.), Formalizing Common Sense: Papers by John McCarthy, 1998, pp. 167–174, originally written in 1982.
[114] J. McCarthy, Elaboration tolerance, in: Working Papers of the Fourth International Symposium on Logical Formalizations of Commonsense Reasoning, Commonsense-1998. Revised and expanded version available at http://www-formal.stanford.edu/jmc/elaboration.html, 1998.
[115] J. McCarthy, Elaboration tolerance, in: Working Papers of the Fourth International Symposium on Logical Formalizations of Commonsense Reasoning, Commonsense-1998, URL http://www-formal.stanford.edu/jmc/elaboration.html, 1998.
[116] J. McCarthy, Formalization of two puzzles involving knowledge, in: V. Lifschitz (Ed.), Formalizing Common Sense: Papers by John McCarthy. Intellect, 1998, pp. 158–166.
[117] J. McCarthy, Mathematical logic in artificial intelligence, in: V. Lifschitz (Ed.), Formalizing Common Sense: Papers by John McCarthy. Intellect, 1998, pp. 237–249, originally published in Daedalus, Winter 1998, pp. 297–311.
[118] J. McCarthy, Actions and other events in situation calculus. in: D. Fensel, F. Giunchiglia, D. McGuinness, M. Williams (Eds.), Proceedings of the Eighth International Conference on Principles of Knowledge Representation and Reasoning (KR-02), 2002, pp. 615–628.
[119] J. McCarthy, Notes on self-awareness, in: Working Papers, DARPA Workshop on Self-Awareness, 2004.
[120] J. McCarthy, The well-designed logical child, Artificial Intelligence 172 (18) (2008) 2003–2014.
[121] J. McCarthy, P.J. Hayes, Some philosophical problems from the standpoint of artificial intelligence, in: B. Meltzer, D. Michie (Eds.), Machine Intelligence, vol. 4, Edinburgh University Press, Edinburgh, 1969, pp. 463–502.
[122] D.V. McDermott, A critique of pure reason, Computational Intelligence 3 (1987) 151–160.
[123] D.V. McDermott, J. Doyle, Non-monotonic logic I, Artificial Intelligence 13 (1–2) (1980) 41–72.
[124] S.A. McIlraith, Integrating actions and state constraints: A closed-form solution to the ramification problem (sometimes), Artificial Intelligence 116 (1–2) (2000) 87–121.
[125] S.A. McIlraith, T.C. Son, Adapting Golog for composition of semantic web services, in: Proceedings of the Eighth International Conference on Principles and Knowledge Representation and Reasoning (KR-02), 2002, pp. 482–496.
[126] R. Miller, L. Morgenstern, The commonsense problem page, http://www-formal.stanford.edu/leora/commonsense, 1997.
[127] R. Miller, M. Shanahan, Narratives in the situation calculus, Journal of Logic and Computation 4 (5) (1994) 513–530.
[128] R. Miller, M. Shanahan, Some alternative formulations of the event calculus, in: A.C. Kakas, F. Sadri (Eds.), Computational Logic: Logic Programming and Beyond, in: Lecture Notes in Computer Science, vol. 2408, Springer, 2002, pp. 452–490.
[129] R.C. Moore, Reasoning about knowledge and action, SRI Technical Report 191, 1980.
[130] R.C. Moore, Semantical considerations on nonmonotonic logic, Artificial Intelligence 25 (1) (1985) 75–94.
[131] L. Morgenstern, Foundations of a logic of knowledge, action, and communication, Ph.D. thesis, New York University, 1988.
[132] L. Morgenstern, A formal theory of multiple agent nonmonotonic reasoning, in: Proceedings of the Eighth National Conference on Artificial Intelligence (AAAI-90), 1990, pp. 538–544.
[133] E.T. Mueller, Event calculus, in: F. van Harmelen, V. Lifschitz, B. Porter (Eds.), Handbook of Knowledge Representation, Elsevier, 2008, pp. 671–708 (Ch. 17).
[134] A. Newell, J. Shaw, H. Simon, Empirical explorations of the logic theory machine, in: Proceedings of the Western Joint Computer Congress, 1956, pp. 218–239.

[135] E.P.D. Pednault, ADL: Exploring the middle ground between STRIPS and the situation calculus, in: Proceedings of the First International Conference on Principles of Knowledge Representation and Reasoning (KR-89), 1989, pp. 324–332.
[136] D. Perlis, Autocircumscription, Artificial Intelligence 36 (2) (1998) 222–236.
[137] J.A. Pinto, Temporal reasoning in the situation calculus, Ph.D. thesis, University of Toronto, Toronto, Canada, 1994.
[138] F. Pirri, The well-designed logical robot: learning and experience from observations to the situation calculus, Artificial Intelligence 175 (1) (2011) 378–415 (this issue).
[139] F. Pirri, R. Reiter, Some contributions to the metatheory of the situation calculus, Journal of the ACM 46 (3) (1999) 325–361.
[140] A. Prior, Diodoran modalities, Philosophical Quarterly 5 (1955) 205–213.
[141] A. Prior, Time and Modality, Oxford University Press, Oxford, 1957.
[142] A. Prior, Past, Present, and Future, Clarendon Press, Oxford, 1967.
[143] R. Reiter, On closed world data bases, in: Logic and Data Bases, 1977, pp. 55–76.
[144] R. Reiter, A logic for default reasoning, Artificial Intelligence 13 (1–2) (1980) 81–132.
[145] R. Reiter, The frame problem in the situation calculus: A simple solution (sometimes) and a completeness result for goal regression, in: V. Lifschitz (Ed.), Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy, Academic Press, 1991, pp. 359–380.
[146] R. Reiter, Knowledge in Action, MIT Press, Cambridge, Massachusetts, 2001.
[147] N. Rescher, Hypothetical Reasoning, North Holland, Amsterdam, 1964.
[148] A. Robinson, Proving theorems as done by man, machine, and logician, in: Summaries of Talks Presented at the Summer Institute for Symbolic Logic 1957, 2nd ed., Institute for Defense Analysis, 1960.
[149] J.A. Robinson, Theorem-proving on the computer, Journal of ACM 10 (2) (1963) 163–174.
[150] J.A. Robinson, A machine-oriented logic based on the resolution principle, Journal of ACM 12 (1965) 23–41.
[151] E. Sandewall, Features and Fluents: The Representation of Knowledge about Dynamical Systems, vol. I, Oxford University Press, Oxford, 1994.
[152] E. Sandewall, Logic based modelling of goal-directed behavior, in: A.G. Cohn, L.K. Schubert, S.C. Shapiro (Eds.), Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning (KR-98), 1998, pp. 304–315.
[153] E. Sandewall, From systems to logic in the early development of nonmonotonic reasoning, Artificial Intelligence 175 (1) (2011) 416–427 (this issue).
[154] R.B. Scherl, H.J. Levesque, The frame problem and knowledge-producing actions, in: Proceedings of the Eleventh National Conference on Artificial Intelligence (AAAI-93), 1993, pp. 689–695.
[155] R.B. Scherl, H.J. Levesque, Knowledge, action and the frame problem, Artificial Intelligence 144 (1–2) (2003) 1–39.
[156] S. Schiffel, M. Thielscher, Fluxplayer: A successful general game player, in: Proceedings of the Twenty-Second Conference on Artificial Intelligence (AAAI-07), 2007, pp. 1191–1196.
[157] L.K. Schubert, Monotonic solution of the frame problem in the situation calculus: An efficient method for worlds with fully specified actions, in: H. Kyburg, R. Loui, G. Carlson (Eds.), Knowledge Representation and Defeasible Reasoning, Kluwer Academic Publishers, Dordrecht, 1990, pp. 23–67.
[158] M. Shanahan, Solving the Frame Problem, MIT Press, Cambridge, Massachusetts, 1997.
[159] M. Shanahan, An abductive event calculus planner, Journal of Logic Programming 44 (1–3) (2000) 207–240.
[160] S. Shapiro, M. Pagnucco, Y. Lespérance, H. Levesque, Iterated belief change in the situation calculus, Artificial Intelligence 175 (1) (2011) 165–192 (this issue).
[161] A. Shirazi, E. Amir, First-order logical filtering, Artificial Intelligence 175 (1) (2011) 193–219 (this issue).
[162] Y. Shoham, Chronological ignorance: Time, nonmonotonicity, necessity and causal theories, in: Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86), 1986, pp. 389–393.
[163] Y. Shoham, A semantical approach to nonmonotic logics, in: Proceedings, Symposium on Logic in Computer Science, 1987, pp. 275–279.
[164] Y. Shoham, Varieties of context, in: V. Lifschitz (Ed.), Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy, Academic Press, 1991, pp. 393–408.
[165] R. Stalnaker, A Note on Nonmonotonic Modal Logics, Department of Philosophy, Cornell University, 1980.
[166] A. Tarski, Logic, Semantics, and Metamathematics, Clarendon, Oxford, 1956.
[167] E. Ternovskaia, ID-logic and the ramification problem for the situation calculus, in: Proceedings of the 14th European Conference on Artificial Intelligence (ECAI), 2000, pp. 563–570.
[168] M. Thielscher, Introduction to the fluent calculus, Electronic Transactions on Artificial Intelligence 2 (1998) 179–192.
[169] M. Thielscher, From situation calculus to fluent calculus: State update axioms as a solution to the inferential frame problem, Artificial Intelligence 111 (1–2) (1999) 277–299.
[170] M. Thielscher, Flux: A logic programming method for reasoning agents, Theory and Practice of Logic Programming (TPLP) 5 (4–5) (2005) 533–565.
[171] M. Thielscher, How (not) to minimize events, in: A.G. Cohn, L.K. Schubert, S.C. Shapiro (Eds.), Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR-1998), 1998, pp. 60–73.
[172] M. Thielscher, A unifying action calculus, Artificial Intelligence 175 (1) (2011) 120–141 (this issue).
[173] P.H. Tu, T.C. Son, M. Gelfond, A.R. Morales, Approximation of action theories and its application to conformant planning, Artificial Intelligence 175 (1) (2011) 79–119 (this issue).
[174] H. Turner, Nonmonotonic causal logic, in: F. van Harmelen, V. Lifschitz, B. Porter (Eds.), Handbook of Knowledge Representation, Elsevier, 2008, pp. 759–776 (Ch. 19).
[175] J. van Benthem, Mccarthy variations in a modal key, Artificial Intelligence 175 (1) (2011) 428–439 (this issue).
[176] G.H. von Wright, Deontic logic, Mind 60 (1951) 1–15.
[177] G.H. von Wright, An Essay in Deontic Logic and the General Theory of Action, North Holland, Amsterdam, 1968.
[178] H. Wang, Toward mechanical mathematics, IBM Journal of Research and Development 4 (1) (1960) 2–22.
[179] A.N. Whitehead, B. Russell, Principia Mathematica, University Press, 1957, originally published 1910–1913.